Lwarp

LaTeX HTML5

The lwarp package

LaTeX to HTML

v0.911 — 2023/02/28

bd@BDTechConcepts.com

Abstract

The lwarp package converts LaTeX to HTML by using LaTeX to process the user's document and directly generate HTML tags. External utility programs are only used for the final conversion of text and images. Math may be represented by SVG images or MathJax. More than 500 LaTeX packages and classes are supported, of which more than 90 also support MathJax.

Documents may be produced by DVI or PDF LaTeX, LuaLaTeX, XeLaTeX; by several CJK engines, classes, and packages; or by customized systems such as perltex and pythontex. A *texlua* script automates compilation, index, glossary, and batch image processing, and also supports *latexmk*. Configuration is semi-automatic at the first manual compile. Support files are self-generated. Print and HTML versions of each document may coexist.

Assistance is provided for HTML import into EPUB conversion software and word processors.

Requirements include the commonly-available Poppler utilities (included with MiKTeX) and Perl. Detailed installation instructions are included for each of the major operating systems and TeX distributions.

A quick-start tutorial is provided, as well as extensive documentation for special cases, a general index, and a troubleshooting index. Automatic error testing is provided for configuration files, package load order, and image generation.

SVG math and many other generated images include LaTeX expressions in the `alt` tags. MathJax may be used with advanced equation numbering under the direct control of lwarp.

Complicated tables are supported, which copy/paste well into LibreOffice Writer.

Supported classes and packages include memoir and koma-script, cleveref, caption, mdframed, siunitx, and many popular packages for tabulars, floats, graphics, theorems, the title page, bibliography, indexing, footnotes, and editorial work, as well as a number of CJK-related classes and packages.

TeX is a self-modifying tokenized macro-expansion language. Since lwarp is written directly in LaTeX, it is able to interpret the document's meaning at a deeper level than external conversions which merely approximate TeX. HTML5 and CSS3 are leveraged to provide advanced features such as booktabs trim, multicolumns, side-by-side minipages, and JavaScript-free navigation.

For a quick-start tutorial, see section 5, Tutorial.

For a list of supported features, see table 2: Supported packages and features.

To update existing projects, see section 1: Updates.

Lwarp is still in development. Changes are likely.

Support TEX development

TEX and related projects:

- are mostly open-sourced and a volunteer effort;
- benefit students, academics, scientists, engineers, and businesses;
- help drive education, public and private research, and commercial activity;
- are used in the fields of mathematics, science, engineering, and humanities;
- are international in reach;
- span decades of development;
- are enduring — many older packages are still actively used and maintained;
- are largely backwards compatible;
- are portable across all the major computing platforms;
- are usable even on older computers and away from internet access;
- are continuing to maintain relevance with modern improvements;
- require no yearly subscription fees;
- and are supported by an active community of knowledgeable volunteers.

Please consider helping by joining and/or contributing to the TEX Users Group, a United States 501(c)(3) tax-exempt charitable organization. Contributions are accepted by credit card, check, or Pay Pal, via the United Way, or by USA or European bank transfer. Membership in TUG supports the development of TEXLive, the major TEX distribution.

Donations may be directed towards individual projects:

TUG Bursary Fund: Assistance for attending annual TUG meetings.
CTAN: The Comprehensive TEX Archive Network — Central storage for TEX.
TeX Development Fund: Support for specific projects.
EduTeX: Teaching and using TEX in schools and universities.
GUST e-foundry fonts: Enhanced for math and additional language groups.
LaTeX Project: Modernizing the LATEX core.
Libre Font Fund: Fonts, tools (FontForge), and distribution (the Open Font Library).
LuaTeX: Combining the PDF TEX engine and the Lua language.
MetaPost: Postscript graphics.
MacTeX: TEX for Mac.
PDF Accessibility: Modern PDF standards.
Other: Additional projects may be specified.

To make a contribution: https://www.tug.org/donate.html

For country-specific TEX users groups: http://tug.org/usergroups.html

For users of MiKTEX: https://miktex.org/donations.html

Contents

List of Figures

List of Tables

1 Updates

The following is a summary of updates to lwarp, highlighting new features and any special changes which must be made due to improvements or modifications in lwarp itself.

For a detailed list of the most recent changes, see the end of the Change History on page 1334.

v0.911: Updated mismath, tcolorbox.

v0.910: Updated fvextra, minted.

v0.909: `\ref` fix.

- Fixed `\ref*`, beamerarticle, lyluatex, realscripts.
- Updated mismath, nicematrix, pablance, pdfpages, simplebnf, tagpdf.

v0.908: Bug fix.

- Fixed obscure cross-reference issue, seen in some citations.

v0.907: Bug fix.

- Fixed SVG images for WINDOWS.

v0.906: Screen readers

- For each `tabular`, add a hidden HTML header cell to convince screen readers that the tables are data not layout. Also hide from the screen reader any final row used only to produce bottom borders.
- Adjusted SVG math for a margin change in *pdfcrop*.
- Added `\Ref`.
- Added docs regarding math in custom environments. See section 8.7.

v0.905: Bug fixes, internal improvements.

- Fixed conflict between cleveref and splitidx.
- Improved coexistence with `\AtEndDocument`.
- acronym: Updated to v1.47, added hyper links.

v0.904a: Fixed missing lwarp-common-mathjax-siunitx package.

v0.904: Added siunitx v3.

- Fixed HTML tags inside non-Latin text.
- MATHJAX now defaults to SVG rendering.
- Added siunitx v3. Updated siunitx-v2. See section 8.7.15 for limitations.
- Updated caption, chemmacros, fbox, hyperref, multicol, wrapfig2.

v0.903: Various updates and improvements.

lwarpmk
- Error if *pdftotext* not available. Ensures that POPPLER programs are installed.

core
- *ps2pdf*: Allow transparency due to recent changes in *ps2pdf*.

⚠ New images
- Due to changes in how automatically-generated SVG image file names are computed, after **`lwarpmk html`** use **`lwarpmk cleanlimages`** a single time, and then **`lwarpmk limages`** to generate the new images.
- Improved back refs.
- Fixed `verbatim*`.
- Various internal updates for recent LaTeX release.

packages
- cuted: Updated to v2.0.
- flushend: Updated to v4.0.
- mathalpha: Updated for v1.14+.
- minted: Updated to v2.6.
- cases: Updated to v3.2.
- siunitx with MathJax: Improved `\per`, `\numlist`, `\SIlist`, comma decimal points.
- Added showlabels, wrapfig2.

v0.902: beamerarticle, footnotes, paragraph tags.

core
- Fixed footnotes inside descriptions, minipages, amsthm, `\nameref`.
- Improved various paragraph tags.

packages
- Improved parnotes, sympytex.
- Added beamerarticle.
- Updated luatexko, xetexko, tagpdf.

MathJax
- Added missing standard international text symbols for MathJax.

v0.901: Tabular columns, float caption CSS, MathJax packages.

core
- Added `warpsvg` to isolate SVG math, as opposed to `warpMathJax`.
- Improved float caption CSS for newer browsers.
- Improved emulation of `\newcolumntype`.
- Added `\HTMLnewcolumntype`. See section 7.6,
- `>{\centering\arraybackslash}`, etc. now sets HTML CSS `text-align`. Also detects `\itshape`, `\bfseries`, and `\bfseries\itshape`. See section 8.10.1.

MathJax
- Now uses MathJax 3.2 packages for centernot, colortbl, gensymb, mathtools, textcomp, upgreek.

packages
- dcolumn: Now works inside a `lateximage`.
- Added mwe.
- Added lltjp-tascmac, which fixed ascmac.

v0.900: Package updates.

core
- Fix for detecting `\usepackage{lwarp}`.

packages
- amsmath: Fixed `alignat` with MathJax.
- changes: Updated to v4.2.1.
- froufrou: Updated to v1.4.0.
- lipsum: Updated to v2.3.

v0.899: Minor updates.

core
- *lwarpmk*: Warns if `\usepackage{lwarp}` is not detected.

packages
- graphics: Added support for `keepaspectratio`.
- keyfloat: Fix: `lw` with `h`.
- multicol: Improved css.

v0.898: Minor updates.

- Fewer underfull `\hbox` warnings.
- wrapfig: Improved integration with keyfloat.

v0.897: siunitx rollback.

docs
- Added a table of file extensions to use with `\includegraphics`. See table 9.

core
- Added tests for additional incompatible packages.

packages
- siunitx: Supports rollback to v2. Does not yet support v3.
- fixme: Improved to work if the user modifies layouts.
- float: Improved integration with newfloat, keyfloat.
- Added centerlastline, decorule, fancypar, froufrou, pbalance.
- Verified works as-is with fnpct.

v0.896: Back references, accessibility.

⚠
- Due to changes in cross referencing, execute **`lwarpmk clean`** before recompiling.
- Increased sectioning nesting stack depth. Error if overflow stack.
- Fixed footnotes at the end of the document, or inside a `description` label.
- Added an error if using braces inside `\usepackage` options.

MathJax
- Fixed footnotes in bracket display math with MathJax.

theorems
- LaTeX theorems, amsthm, ntheorem, theorem: Print theorem footnotes following theorems.

accessibility
- Added HTML `<main>` element to each page.
- Added ARIA `math` role to SVG math images, and `note` role to margin notes, footnotes, etc.

packages
- Improved citation backreferences for various packages.
- chemfig: Updated to v1.6a.
- bigdelim: Updated to v2.8.
- xetexko: Updated to v3.1.

- hyperxmp: Fix: Accept and discard additional keys.
- hyperref: Fix: Added `\*autorefname` macros.
- biblatex: Fix: Back references.
- tocloft: Fix: `\cftpagenumbersoff`, `\cftpagenumberson`.
- threeparttablex: Fix: `\TPTL@tnotex`.
- amsthm: Fix: Footnotes inside environment optional argument.
- listings: Fixed labels. Accepts but ignores escapes w/o error.
- pdflscape: Fix: Added `landscape` environment.
- Added ccicons, classicthesis, orcidlink.
- Added enotez.
- Verified support for doi, doipubmed.

v0.895: Vector packages, greatly improved MathJax for siunitx.

core
- Fixed quotes in HTML tags while using old font packages with XeLaTeX and LuaLaTeX.

MathJax
- Added `\ifblank` and `\ifstrequal` to MathJax emulation.

packages
- multirow: Allow `\par` per v2.7.
- acro: Updated to v3.5.
- fancyhdr: Updated to v4.0.
- changes: Updated to v4.0.1.
- epsfig, rotating: Now work inside `lateximage`.
- amscdx: Verified to work with SVG math. Warning added about use with MathJax.
- Added MathJax emulation for isomath, mattens, maybemath, skmath, tensor.
- Improved MathJax emulation for siunitx `\ang`, `\num`, `\SI`.
- Added epsf, impnattypo, isotope, lpic, luavlna, mdwmath, pinlabel, rlepsf, tikz-imagelabels, xevlna.
- Verified to work as-is: tensind.

v0.894: MathJax additions and improvements.

MathJax
- Improved warning message for enabling SVG graphics for select math expressions while using MathJax.
- Accept and ignore a star for `\hspace`.
- Ignores `\arabic`, `\number`, `\noalign`.

packages
- Added MathJax emulation for backnaur, colortbl, nicematrix.
- booktabs: MathJax emulation now absorbs and discards trim.
- menukeys: Updated to v1.6.1.

v0.893: Minor fixes, more packages.

MathJax
- Added MathJax emulation for `\mathnormal`.

packages
- Fixed pstricks `pspicture*`.
- Fixed tikz font macros.
- braket: Now uses the MathJax extension.

- Added esvect, fixmath, keystroke, mathastext, menukeys, picinpar, plimsoll, repltext, selectp, seqsplit, simplebnf, statistics, swfigure.
- Added MathJax emulation for mathspec.
- Verified to work as-is for apxproof, syntaxdi, venndiagram.

v0.892: minted, fvextra, MathJax `\left`/`\right`.

MathJax
- fourier, libertinust1math, newpxmath, newtxmath, newtxsf, unicode-math: Added MathJax `\left`/`\right` support for additional delimiters.

packages
- textpos: Updated to v1.10.
- xcolor: Fixed optional args for `\fcolorbox` and related.
- Added fvextra, minted.

v0.891: MathJax additions and improvements.

core
- Now displays inline `\verb` text as `\texttt`.
- Fixed alltt and verbatims with LaTeX lists.
- Now generates an error if nested each of `warpHTML`, `warpprint`, `warpMathJax` inside itself.

MathJax
- Added MathJax *textmacros* extension, allowing formatting inside `\text`.

packages
- biblatex, hyperref: Added back page references.
- fancyvrb: Fixed `BVerbatim` with a label.
- listings: Fixed MathJax with captions, improved HTML sanitation.
- babel-french: Fixed `\texorpdfstring` conflict.
- Now honors Greek package options for mathdesign, mathpazo, mathptmx, newpxmath, newtxmath.
- Improved MathJax for colonequals, mathdesign, mathdots, mathfixs, mathtools, multiobjective, nicefrac, shuffle, units.
- unicode-math: Added Greek macros, as well as macros for the first several categories listed in **`texdoc unimath-symbols`**. Improved symbol shape macros with Greek. Improved documentation.
- Added bussproofs, cmbright, fourier, kpfonts, kpfonts-otf, libertinust1math, scalerel, txgreeks.

v0.89: Additional MathJax support.

core
- Adapted to upcoming LaTeX kernel changes.
- Allows load of amsmath before lwarp.

lwarpmk
- Also removes `*.bbl` when cleaning aux files.

MathJax
- MathJax: Neutralized `\protect`, `\mathcode` and related, ligatures. Fixed nested environments.

packages
- caption: Updated for v3.5, fix for label sep.
- thmtools: Updated for v0.72. Fixed `swapnumber`, `margin`.
- Improved MathJax for centernot, mathtools, mismath, SIunits, siunitx, statmath.
- Added MathJax emulation for accents, hepunits, hhtensor, mathalpha, mathdesign, mathpazo, mathptmx, mleftright, newpxmath, newtxmath, newtxsf, pxfonts, shuffle, txfonts, upgreek, ushort.
- Verified to work as-is: authoraftertitle.

v0.88: Indexing, boxing, theorems.

- Now has programmed support for more than 500 packages and classes, of which more than 60 also support MathJax.

core
- Fixed: `\ref*`, and also added MathJax emulation.
- If starting a new paragraph, `\hrulefill` creates a `<div>` with a thin horizontal line across the page. Use instead of `\hrule`.
- Fixed: Use `\chaptername` where appropriate.
- Fixed: Inline links causing extraneous paragraphs.

lwarpmk
- Added **`lwarpmk -v`** to print the version number.

indexing
- Added the `IndexRef` option to control the display of index entries. See section 7.5.
- Added `\IndexPageSeparator` and `\IndexRangeSeparator` for custom index styles.
- Added support for gindex, *xindex*.
- Verified to work as-is with varindex.

packages
- cleveref, varioref: Fix for starred macros.
- varioref: Removed page-related text from HTML output.
- xfakebold: Updated to v0.08, using pdfrender.
- caption, scrextend: Fixed `\caption*`.
- Added fbox, shadethm, tcolorbox, termcal, thmbox, thmtools.

v0.87: MathJax, bibliography packages.

core
- Added boolean `FixSmallCaps` for fonts which render small caps as all caps.
- Fixed `\bibliography` to use the HTML version's `.bbl` file. Previously the HTML bibliography relied on the print version's `.bbl`, thus would fail if the print document had not yet been created.

MathJax
- Added `\ifstar` and `\ifnextchar` to MathJax, and removed `\DeclareIfstar`. See section 8.7.7.

⚠ Removed `\DeclareIfstar`

packages
- physics: Now supports the MathJax v3 extension.
- mathtools: Improved `\underbrakcet`, `\overbracket` for MathJax.
- nccmath: Improved `\underrel` for MathJax.
- mhchem: Now supports the MathJax v3 extension for `\ce` inside math.
- cancel: Now supports the MathJax v3 extension.
- embrac: Neutralized kerning for improved HTML conversion.
- Added citeref, drftcite, jurabib, multibib, splitbib.
- Verified to work as-is with bibtopic, collref, mciteplus.

v0.86: MathJax major updates.

core
- Fixed: Filename if named files with `*`, parens, period in section name.
- Fixed: Labels in `eqnarray`, `lateximage`.

MathJax
- Updated to MathJax v3. New repository.
- Fixed forward references for MathJax.

- Improved MathJax equation number formatting, now compatible with amsmath `\numberwithin` for chapters, sections, subsections, as well as amsmath `subequations`. See section 8.7.7.
- Added `\DeclareIfstar` to define starred TeX macros in MathJax. See section 8.7.7.
- Generates an error if `\MathJaxFilename` file does not exist.

packages
- mathtools, nccmath, physics: Added starred macros for MathJax.
- nccmath: Fixed `\nr`, `\displaybreak` for MathJax.
- xcolor: Fixed `\textcolor` with babel-french.

v0.85: fontspec

packages
- fontspec: Fixed core font change macros for world languages.

⚠ acro formats
- acro: Due to v3 changes, when defining acronym formats, use `\textbf` instead of `\bfseries`, etc.
- Fixed idxlayout, mathtools, titlesec, url.

v0.84: Previous/next page links, numerous fixes.

docs
- Added documentation of `BlockClass` and `\InlineClass` for CSS `<div>`s and `<span>`s. See section 7.8.
- Added `\LinkPrevious`, `\LinkNext` page links. See section 7.6.

⚠ home page footer changed
- Added `\FirstPageBottom`. Home page no longer shares `\PageBottom`. See section 7.6.

core
- Improved coexistence with comment, support for nested environments.
- No longer requires but still supports the caption package.
- Improved filenames and HTML titles when using special characters.

⚠
- Change: Append `-0` to section named Index previously _index to distinguish from `index.html`
- Fixed style tags for `\multicolumn`, `\multirow`.
- Fixed spacing in `tabbing`.
- Fixed `lateximage` for: `quote`, `quotation`, `verse`, `center`, `flushleft`, `flushright`, `<par>` tags, packages verbatim, alltt, epigraph.
- Fixed textcomp due to integration into LaTeX kernel.
- Fixed `\itshape`, etc. Adapted to LaTeX fontaxes integration.
- Fixed `\@fnsymbol`.
- Warns about section names with dollar-delimited math.
- Warns about a `<span>` containing a float, caption, section, mdframed, or other `<div>` object.
- Only warn about XeTeX logo and graphics if actually used `\Xe`.

lwarpmk
- **`lwarpmk clean`** also removes `comment_*.cut`.

packages
- scrextend, scrartcl, scrbook: Added `\titlehead`, `\subject`, `\subtitle`, `\publishers`.
- titling: Fixed `\printthanks`.
- memoir, abstract: Fixed for updated memoir.
- memoir: Fixed `\newcomment`, pagenotes, crossreferences. Fixed setting a recursive name.

- Fixed or improved: amsthm, backref, biblatex, fixme, nfssext-cfr, ntheorem, parcolumns, realscripts, rotfloat, titling.
- Added boxedminipage, renamed from boxedminipage2e per author.
- Verified to work as-is with mcite.

v0.83: memoir fixes.

packages
- memoir: Various fixes and updates.
- physunits: Updated to v1.0.4.

v0.82: MathJax notes, xpinyin improvements, various updates.

MathJax
- Improved footnotes with MathJax.
- Added MathJax emulation for endnotes, marginnote, nccfoots, pagenote, parnotes, sidenotes.

packages
- xpinyin: Added pinyin with modern HTML.
- luatexko: Added `\dotemph`, `\ruby`, `\uline`, etc.
- soul: Fixed `\<`.
- chemfig: Updated to v1.5.
- draftwatermark: Updated to v2.0.
- ulem: Fixed: `\dashuline`.
- amsmath: Fixed: `\intertext` with MathJax.
- endnotes: Fixed: Marks in print mode.
- tocvsec2, tableof: Verified to work as-is.
- Added etoc (nullified).

v0.81: MathJax speedup and additional emulations.

core
- Improved warning regarding SVG math sizing / baselines and graphics / graphicx. See section 8.7.

MathJax
- Improved MathJax emulation processing speed.
- Added MathJax emulation for accsupp, axessibiltiy, colonequals, decimal, dotlessi, econometrics, engtlc, multiobjective, physunits, SIunits, stackrel, statmath.

packages
- axessibility: Updated to 2020/01/08 version.
- gridset: Updated to v0.3.
- SIunits: Fixed for math mode.
- Added DotArrow, nolbreaks, luamplib, returntogrid, statex2, tagpdf.
- Verified to work as-is with icomma, mathpunctspace, textualicomma.

v0.80: MathJax, biblatex.

MathJax
- Added docs and warning/info messages re: avoiding slow MathJax compilation. See section 8.7.7, Customizing MathJax.
- Added MathJax emulation for accessibility, autobreak, centernot, extarrows, fouridx, gensymb, leftidx, mathcomp, mathdots, mathfixs, mismath, nccmath, noitcrul, pdfcomment, relsize, rmathbr, subsupscripts, xfrac.
- Improved MathJax emulation for unicode-math.

packages
- biblatex, url: Now create hyperlinks.

- amsmath: Fix to center starred environments.
- xcolor, graphics: Made more macros robust.
- colortbl: Fix: Rule color in a `lateximage`.
- chemmacros: Updated to v5.10.
- Added fewerfloatpages, ghsystem, hhline, mismath, nccmath.

v0.79: MathJax, nested `tabular`.

MathJax
- Added or improved MathJax emulation for amsmath, ar, arydshln, bm, bigdelim, bigstrut, booktabs, braket, mathtools, multirow, physics, siunitx, slashed, unicode-math, xfakebold.
- Warn if using certain packages not supported by MathJax.

core
- `tabular`: Now may be nested.
- `minipage`, `\parbox`, `fminipage`, `\makebox`, `\framebox`: Fix: Adjust for virtual page size.
- Uses new iftex.

packages
- graphicx: Fix: Negative angles.
- caption: Fix: `\captionlistentry` with longtable.
- multirow: Fix: Centered vertical alignment.
- siunitx: Fix: `\square`, `\cubed`.
- booktabs: Fix: memoir with `lateximage`.
- babel and polyglossia: Added troubleshooting warnings.
- fontawesome, fontawesome5: Supports text color and size.
- transparent: Fix: `lateximages`.
- epigraph: Updated to v1.5e.
- xurl: Updated to v0.08.
- subcaption: Fixed with memoir.
- floatrow: Fix: `\linewidth`. No longer require float, graphics.
- floatflt, wrapfig, niceframe: Fix: Adjust for virtual page size.
- Added widetable, witharrows, steinmetz.
- Added awesomebox, catoptions.
- Added svg, supports svg-extract.
- Added parcolumns, pdfcolparcolumns,
- Added parallel, pdfcolparallel.
- Added pdfcol, pdfcolfoot, pdfcolmk.

v0.78: Fixes for support files, `alt` tags, hyperlinks, and the 2019/10 LaTeX release.

docs
- Docs: Improved documentation regarding package options. See section 8.1.
- Fix to overwrite existing support files using new `filecontents` environment.

packages
- breqn: Previously broken by the 2019/10 LaTeX update, but now working again.
- graphics: Fix for `\includegraphics` `alt` tags.
- babel-french: Fix for hyperlinks.
- media9, movie15, multimedia: Fix for the 2019/10 LaTeX update.
- accessibility: Added.

v0.77: Updates to fix recently-broken packages.

- booktabs: Updated to v1.6180339.
- chemformula: Updated to v4.15.

v0.76: MathJax, updates for LaTeX 2019/10 release.

docs
- Docs: Expanded documentation regarding the use of multiple projects in the same directory. See section 5.17.

MathJax
- MathJax: Updated to v2.7.6.

packages
- xr: Updated to v5.05.
- xr-hyper: Updated to v6.1.
- Verified works as-is with xcite.
- acro: Updated to v2.10.

⚠ broken
- Currently broken in print mode by the 2019/10 LaTeX update, and waiting for fixes: breqn, grffile, multimedia, movie15.

v0.75: keyfloat, wrapfig

- `\minipage`: Fix for `\linewidth`.

packages
- keyfloat: Improved color control.
- wrapfig: Fix for `\linewidth`.

v0.74: Docs, SVG math, *lwarpmk*, HTML `alt` and `title` text, lyluatex

docs
- Added to the tutorial the section What next?. See section 5.19.
- Added documentation about localization options. See section 7.1.
- Added documentation about accessibility options. See section 7.2.

HTML `alt` text
- Renamed and updated HTML `alt` text macros:

⚠ changed names

Old	New
(hard coded as "image")	`\ImageAltText`
`\mathimagename`	`\MathImageAltText`
`\packagediagramname`	`\PackageDiagramAltText`

- Added `\ImageAltText` for the default HTML `alt` text for an image. See section 7.6.
- Added `\ThisAltText`, which may be used to assign a one-time HTML `alt` tag to the very next image generated by lwarp, such as a `lateximage`, `picture`, `tikzpicture`, an image generated by various chemistry or engineering packages, or an SVG math image. This macro also adds a `title` tag to a reference or hyperlink. See section 7.6.

SVG math
- Adjusted `\LateximageFontScale` default from .75 to 1.
- Fix: Font control for SVG math.

misc
- Fix: Ignores negative `\hspace`.
- Warning if `SideTOCDepth` < `FileDepth`.

lwarpmk
- *lwarpmk*: **`lwarpmk clean`** removes additional files.
- *lwarpmk*: **`lwarpmk epstopdf`** and **`lwarpmk pdftosvg`** now honor directories.

packages
- lyluatex: Split images by `system` or per `fullpage`, improved margins and scaling.
- Tested to work as-is with mathspec, unicode-math.

v0.73: `\include`, memoir, koma-script, caption, xy, datatool, music scores.

- Fix for `\include`.
- Warning for a `tabular` inside a `<span>`.
- `\color`: Added HTML support for rules and frames, but not inline text. Use `\textcolor` if possible.
- Improved many HTML tags, reducing *tidy* warnings. See Change History.

packages

- memoir: Fixes for `\frontmatter*` and `\mainmatter*`. Added `\book`.
- koma-script: Fix for starred captions in the TOC.
- caption: Fix for starred captions.
- datatool: Added pie, bar, and plot charts.
- threeparttable: Added `measuredfigure`.
- intopdf: Updated to v0.2.1.
- tocdata: Updated to v2.03.
- quotchap: Updated to v1.2.
- versonotes: Updated to v0.4.
- backnaur: Now uses SVG images. Updated to v3.1.
- xy: Fix for `\xybox`, improved xy, also now compatible with qcircuit.
- fancyvrb: Fix for label HTML tags.
- Added stackengine.

music

- Added lyluatex. (Music scores.)
- musicography: Updated to 2019/05/28. Added support for `lateximages`.

v0.72: Font control, `\multicolumn`, xr and xr-hyper.

⚠ images

- Due to internal changes, images for inline SVG math and `lateximages` will have new hash values, and will have to be regenerated using

  Enter ⇒ **`lwarpmk cleanlimages`**

  and

  Enter ⇒ **`lwarpmk limages`**
- Docs: Color-codes package names in the table of supported packages and features, table 2, according to each package's level of support by lwarp.
- `\multicolumn`: Fix for paragraph columns.

packages

- xr, xr-hyper: Fixes for references, `\externaldocument`.
- soulutf8: Fix: Loads soul for emulation.
- boxedminipage2e: Added support for `lateximages`.
- zhlineskip: Updated to v1.0e.
- Added fontaxes, slantsc, tabfigures.
- Added nfssext-cfr, thus supporting cfr-lm and several other font packages.
- Added backnaur, hypbmsec, minibox, pdfcrypt, shapepar.

v0.71: Error handling, multimedia, `tabular`.

- `tabular`: Added support for '`*`' columns. Fix for paragraph tags.
- `quotation`: Fix for HTML tag.
- Docs: Added a section about error conditions tested by lwarp. See section 13.1.
- *lwarpmk*: If file `lwarpmk.conf` is an older version, or the incorrect operating system, displays the print command to use to recompile.

packages
- chemfig: Updated for v1.4.
- endfloat: Updated for v2.7.
- textpos: Updated for v1.9.1.

multimedia
- Added media9, movie15, multimedia.

v0.70: Error handling, MathJax, mathtools.

- Error handling for "Label(s) changed." Refuses to **`lwarpmk limages`** until recompile first.
- Fix: If Computer Modern font is used, ensures cm-super or lmodern is used.
- Fixes for `\makebox`.
- Fixes for `\parbox` inside a `<span>`.
- MathJax: Updated to v2.7.5. Loads the `autoload-all.js` extension. Added `\MathJaxFilename` to select custom scripts.

packages
- textcomp, xunicode: Fix for `\textinterrobang`.
- mhchem: Works with MathJax. See section 410.
- changes: Updated to v3.1.2.
- Added autonum, changelayout, inputtrc, mathtools, metalogox.

v0.69: Error handling, many fixes, improved keyfloat / tocdata.

- Fix for HTML corruption of `lateximage` displays.
- `\makebox`, `\framebox`: Fix for (⟨*width,height*⟩) arguments.
- `fminipage`: Honors `\minipagefullwidth`.

packages
- array, longtable: Fix for `\tabularnewline`.
- tabularx, tabulary: Fix to require the array package.
- supertabular, xtab: Fix to clear caption after use.
- graphics: Added a warning if used the `\includegraphics scale` option.
- multirow: Added an error if didn't use `\mrowcell` or `\mcolrowcell` when using `\multirow` or `\multicolumnrow`.
- keyfloat: Updated for v2.00, additional improvements.
- Added ctable, eqlist, eqparbox, ftcap, listliketab, minitoc, tocdata, topcapt.

v0.68: Error handling, tabulars, footnotes.

lwarpmk
- *lwarpmk*: Improved error handling for image generation if compile was incomplete.
- `tabular`: Fix for `\warpprintonly`.

packages
- longtable: Improved flexibility for `\endhead`, etc. Improved error reporting if `\endhead`, etc. incorrect for lwarp.

- threeparttable: Fix for caption type.
- hyperref: Fix for options with braces.
- morefloats: Fix to be loaded early for print output.
- listings: Updated for v1.7.
- Added bigfoot, fnpara, footnotebackref, manyfoot, tablefootnote, threeparttablex.
- Added layouts, niceframe, perpage, showtags.
- Prevented alg, algorithmic, pdfcprot, fncylab.

v0.67: Filename generation, symbol fonts.

docs
- Documentation fix for `<project>-images`, `<project>-images.txt`.
- Added discussion regarding section names. See section 8.4.

filenames
- Added `\FilenameNullify` and `\FilenameSimplify` for filename generation. See section 8.4.
- Core, textcomp, xunicode: Nullified additional symbols during filename generation.

packages
- color: Fix for version number warnings.
- Added academicons, bbding, dingbat, eurosym, fontawesome, fontawesome5, marvosym, pifont, typicons.
- Added changes, easyReview, fitbox, foreign, gloss, karnaugh-map, multicap, nomencl, notes, struktex, umoline, xfakebold.
- Tested to work as-is with askmaps, curves, euro, karnaughmap, tikz-karnaugh.

v0.66: xr, multiple projects, image names/directory, HTML formatting

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **`pdflatex filename.tex`** or similar, after which *lwarpmk* may then be used with the new configuration files.

`lateximage`
- Adds options `ImagesDirectory` and `ImagesName` to assign directory and name prefixes for `lateximage` images. The new defaults include the jobname, allowing the image directories for multiple projects to coexist.

⚠ existing projects
- To reuse existing `lateximage` directories, add lwarp options

```
\usepackage[
  ImagesDirectory={lateximages},
  ImagesName={lateximage-}
]{lwarp}
```

  If not reused, the existing `lateximages` directory and `lateximages.txt` file may be removed.

filenames
- Added `\FilenameLimit` to control the maximum length of the filenames generated by lwarp.

⚠ Possible filename changes
- Improved filename generation when special characters or macros are used in section names.

Windows
- Fix for **`lwarpmk cleanlimages`** with Windows.

floats
- Fixes for floats in the home page.

lists, table notes
- Improved CSS for definition lists, table notes.

`tabular`
- `tabular`: Fixes for `\par` in column specifier, minipage inside `tabular`.

indexing
- Indexing: Fix for a long line of multiple entries.

`minipage`
- `\minipagefullwidth`: Fix for global changes.

- Added \UseMinipageWidths and \IgnoreMinipageWidths. See section 8.3.3.

colors
- Improved \fbox, \fboxBlock, \fminipage to use current text color.

HTML
- Improved HTML output formatting.

docs
- Added discussion regarding invalid HTML. See section 8.1.1.
- Added discussion regarding math in section names, \imagegraphics scale option. See section 6.
- Added discussion regarding international languages in section names. See section 8.14.

packages
- caption: Fix for options clash.
- xr, xr-hyper: Now compatible.
- subcaption: Improved horizontal spacing.
- multicol: Fix for minipage inside multicols.
- multicolrule: Updated for v1.2.
- tocbasic: Minor update.
- acronym: Fix for acronym in float caption.
- kotexutf: Patch with *pdflatex* and new lwarp labels.
- extramarks, fancyhdr: Updated for v3.10.
- memoir: Added docs regarding version numbers. See section 8.13.
- zref: No longer required.
- Added ar, ed, indentfirst, nameauth, truncate.
- Verified to work as-is with changelog.
- Prevented colortab, epsf, hyper, picinpar, picins, sistyle, ucs.

v0.65: CSS layout, alt tags, Japanese.

page layout
- Moved the sideTOC to the left side, allowing improved CSS for margin notes.
- Improved page layout CSS.

image alt tags
- graphicx \includegraphics: Added the alt key to assign an alt tag to an image. Default is "image", assigned to pass validation.

duplicate HTML files
- Detects and causes an error if duplicate HTML file names are generated, caused by identical or similar sectioning names.

fixes
- Fix for tabular*.
- Fix for tabular border colors.
- Fixes \quad, \enskip, and figure captions to pass validation.

Japanese
- Added ltj* classes, bounddvi, gentombow, lltjext, plarydshln, plext, plextarydshln, plextcolortbl, pxatbegshi, pxeveryshi, pxftnright, pxjahyper, tascmac.
- Verified to work with plarray, plautopatch, plextarray, plextdelarray, pxgentombow, plsiunitx, pxpdfpages, pxpgfrcs, pxpgfmark.

packages
- Added support for fontspec \textsi and \sishape.
- Added multicol's \docolaction.
- Added embrac, footnoterange, multicolrule, versonotes.

v0.64: Koma-Script, Japanese, Chinese.

Japanese
- Added utarticle and related classes.
- Improved ujarticle and related classes.

Chinese
- Fix for biblatex with CTEX and other classes.

Koma-Script
- Fixes for scrlayer, scrlayer-scrpage.

packages
- addlines: Updated to v0.3.
- Added bsheaders, gmeometric, marginal, rmpage, scrpage2.

v0.63: mdframed, Chinese, Japanese, Korean

localization
- Added `\linkhomename`: A user-definable name for the **Home** link.
- Documented `\sidetocname`: A user-definable name for the sideTOC.

fixes
- Fix: `\LinkHome` for print output.

optimizations
- Moved package load checks to the lwarp core to reduce the number of `lwarp-*` files.

packages
- mdframed: Fix with amsthm, improved titles and font control. Improved rule widths.

Chinese
- Fixes for xeCJK.
- Added xpinyin, zhlineskip.
- Verified to work with cjkpunct, upzhkinsoku, zhspacing.

Japanese
- Verified to work with zxjatype, luatexja, luatexja-fontspec.
- Added bxjsarticle and related classes.
- Added ltjsarticle and related classes.
- Added pLATEX, upLATEX, ujarticle and related classes.
- Prevented utarticle and related classes.
- Prevented bxcjkatype.

Korean
- Verified to work with kotex, xetexko, luatexko.

v0.62: MiKTEX docs, HTML title, CTEX, xeCJK, bitpattern.

docs
- Docs: Setting a UTF-8 locale. See section 9.9.

MiKTEX
- MiKTEX: Docs for *MiKTeX Console* and miktex-poppler-bin.

HTML <title>
- HTML subpage titles: Added `\HTMLTitleBeforeSection` and `\HTMLTitleAfterSection` to select whether the HTML <title> displays the website name before or after the section name. See section 7.6.

fixes
- Fix for package options handling.
- Fixes for horizontal white space between `fminipage`, `fcolorminipage`, `colorboxBlock`, `fcolorboxBlock`.
- Logos: Fix for XƎTEX logo, improved CSS, made robust, improved search-engine optimization.
- `\\[$1]`: Additional HTML <br> if $1 > 0 pt.
- Fixes for `\includgraphics` filename, and with `FormatWP`.
- Fix: CSS for `\textup`.
- Fix: Added `\slshape`.

Chinese
- Added ctex package and related classes, xeCJK.
- Prevented CJK, CJKutf8 unless xeCJK, ctex are used.

packages
- chemfig: Docs for new macro `\polymerdelim`.
- asymptote: Docs for compilation.
- chngpage: Fix to load lwarp-changepage.
- algorithm2e: Fix with non-book classes.
- register: Updated to v1.8.
- nicefrac: Improved font control and CSS, honors `nice` and `ugly`.
- units: Improved font control and CSS, honors `tight` and `loose`.
- xfrac: Improved CSS.
- textcomp and xunicode: Fix conflicts with `\textcircled`.
- ulem: Improved compatibility with CJKulem, `lateximage`.
- MathJax and siunitx: Removed inoperable extension.
- Added bitpattern, pdfcomment, pdfmarginpar, tram, unitsdef, xechangebar.
- Added musicography, octave, semantic-markup.
- Added 2in1, flippdf, notespages, rviewport, twoup.

v0.61: Custom compilation, EPS-related packages, documentation, indexes.

docs
- Split index into multiple indexes.
- Improved documentation regarding font selection. See section 7.4.
- Added documentation regarding debugging options. See section 35.
- Added documentation regarding HTML entities inside program listings. See section 8.2.1.

custom compiling
- Added options to specify the shell commands to execute for **`lwarpmk print`** and **`lwarpmk html`**, allowing the use of lwarp with perltex, pythontex, etc. If not specified, these are set automatically depending on the LaTeX engine, `--shell-escape`, and lwarp options. See section 9.

⚠ changed names
- Changed macro names to match `\displaymathother`, `\displaymathnormal`:

Old	New
`\StartDynamicMath`	`\inlinemathother`
`\StopDynamicMath`	`\inlinemathnormal`

fixes
- Fix: Paragraph tags in a `tabular`.
- Fix: supertabular and xtab captions.
- Fix: DVI LaTeX `\includegraphics` EPS images.
- Fix: newfloat lists.
- Fix: CSS footnotes text align, minipage tabular and footnote margins.

packages
- Added epsfig, psfrag, psfragx, pstool.
- Added copyrightbox, pdfprivacy, thinsp, threadcol, uspace.
- Added chkfloat, cmdtrack, dprogress, lua-visual-debug, refcheck, srcltx, srctex, vpe, xbmks.

v0.60: Fixes for longtable, listings.

fixes
- longtable, etc.: Fixes for slowdown and memory management for very long tables.
- listings: Fix for HTML entities, and also when used inside a list.
- diagbox: Fix for incorrect HTML par tags.

packages
- Added 2up, booklet.
- Added bophook, draftfigure, fullminipage, grid-system, layaureo.
- Added leading, widows-and-orphans.
- Added fancytabs, thumb, thumbs.

v0.59: DVI *latex*, MATHJAX, asymptote, pdftricks and pstricks, epstopdf, brqen.

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **`pdflatex filename.tex`** or similar, after which *lwarpmk* may then be used with the new configuration files.

lwarpmk
- Added an error if `lwarpmk.conf`'s format has changed and the document must be recompiled.
- Added a warning if the `lwarpmk.conf` configuration file appears to be for the wrong operating system, in case files are transferred between systems.
- Added
  ```
  lwarpmk epstopdf <list-of-EPS-files>
  ```
  to quickly convert a document's EPS images to PDF or SVG. See section 8.8.

DVI *latex*
- Added support for DVI *latex*. See section 7.5.

latexmk
- Fix for `--shell-escape` with *latexmk*.

math
- Updated MATHJAX script to v2.7.4.
- Fix: MATHJAX chapter number removed from non-numeric tagged equations.
- Added MATHJAX support for nicefrac, units.
- Fix for `\[` and `\]` with `\displaymathnormal`.

images
- Fix for `\includegraphics` filename expansion.
- `\includegraphics` now works with `.pdf` and `.eps` filename extensions.

packages
- Moved amsmath out of the lwarp core.
- Fix for chemformula `\NMR`.
- Added asymptote, pdftricks, pstricks, pst-eps.
- Added breqn, SIunits.
- Added bxpapersize, canoniclayout, draftcopy, fnbreak, nccfancyhdr.
- Added accsupp, axessibility.
- Added xunicode.
- Improved and now supports epstopdf.
- Tested to work as-is: eepic, sepfootnotes.

docs
- Added information about setting up a development version of lwarp.

v0.58: Extensive improvements in indexing, glossaries. Adds PDF-inclusion packages.

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **`pdflatex filename.tex`** or similar, after which *lwarpmk* may then be used with the new configuration files.

lwarpmk
- *lwarpmk*: Added the `-p` option to specify the project name.

glossaries
- *lwarpmk*: Now uses *makeglossaries* for glossary generation, allowing the processing of multiple glossaries at once.

- Added lwarp option `GlossaryCmd` to specify the shell command used by **`lwarpmk printglossary`** and **`lwarpmk htmlglossary`**. Defaults to **`makeglossaries`**.

index and glossary

- Docs: Extra indexing options. See section 8.6.14.
- Added support for *makeindex*. (Previously supported only *xindy*.) Also added indexing packages listed below.
- Added lwarp options `PrintIndexCmd`, `HTMLIndexCmd`, and `LatexmkIndexCmd` to specify shell commands used by **`lwarpmk printindex`**, **`lwarpmk htmlindex`**, and *latexmk*. May be preset with the `makeindex` or `xindy` lwarp options. See section 7.5.
- Added lwarp options `makeindex` and `xindy` to set `PrintIndexCmd`, `HTMLIndexCmd`, and `LatexmkIndexCmd` to sensible values for a typical single index. See section 7.5.
- Added lwarp option `makeindexStyle` to tell *lwarpmk* to use a custom style instead of `lwarp.ist`. See section 8.6.20.
- Fix for index entries with `\see`, `\seealso`, `\emph`, `\textbf`, etc.

misc. fixes

- Replaced each `\csuse` with `\@nameuse` for improved error detection.
- Additional internal print/HTML macro selection improvements.
- Fix: `\printindex` finishes pending `\index` writes first.

packages

- Fixes for memoir: makeidx, ccaption, multiple indexes, `\specialindex`.
- Fixes for komascript: Indexing improvements.
- Added imakeidx, index, repeatindex, splitidx.
- Added attachfile, attachfile2, intopdf, pdfpages, pdfx.
- Added cases.
- Tested to work as-is: notes2bib, hvindex.

v0.57: algorithm2e, float styles, tabular packages, internal improvements.

MathJax

- Added support for MathJax equations with `\footnote`, `\footnotemark`.

math macros

- Added `\StartDefiningMath` and `\StopDefiningMath` for use when defining macros in the preamble which contain $. See section 8.7.9.

dynamic math

- Added `\inlinemathother` and `\inlinemathnormal` to delimit math expressions which depend on a variable condition such as a counter. Such expressions will not be hashed for reuse, and will be converted to SVG math images even when MathJax is enabled. See section 8.7.10.

⚠ new name

- Renamed `\EndDefiningTabulars` to `\StopDefiningTabulars`.

`lateximage alt` tags

- Improved localization for `lateximage` HTML `alt` tags. For SVG math images, the `alt` tag under some conditions will be set to `\MathImageAltText`, which defaults to `math image`. For packages, the `alt` tag is set using the package name followed by `\PackageDiagramAltText`, which defaults to `diagram`. Ex:

```
(-xy- diagram)
```

  See section 7.6.
- Fix: Improved print/HTML macro selection.

misc. fixes

- Fix: `\href` text catcodes.
- Fix: `\subref` text.

- Fixes: Colored `\rule` and `\boxframe`.

packages
- float, rotfloat: Adds support for float styles `ruled` and `boxed`.
- float: Fix: Do not create `\l@<type>` until `\listof` is used.
- marginnote: Fix: Long optional argument.
- ellipsis: Adds `\midwordellipsis`.
- breakurl: Fix for text catcodes.
- Added algorithm2e, register, ltablex, xltabular, xellipsis, trimclip, errata, vowel, xpiano.
- Prevents glossary.
- Tested to work as-is with gauss, phonrule, piano, SIunits, tikzcodeblocks.

v0.56: Shell escape, tabular packages.

lwarpmk
- Added
  ```
  lwarpmk pdftosvg <list-of-PDF-files>
  ```
  to quickly convert a document's PDF images to SVG, for use with HTML. See section 8.8.
- Added support for `--shell-escape`. See section 7.3.

`tabular`
- Added support for array `w` and `W` columns.
- Fix: `\multicolumn` parameter handling.
- Added support for double `\hlines`, `\midrules`, and vertical rules.
- Added support for arydshln dashed lines with HTML `tabular`, but reverts to plain rules for `lateximage` and SVG math `array`.

misc. fixes
- Fix: `\thinspace`.
- Fix: paralist compact environments.

packages
- Added parnotes, quoting, lua-check-hyphen, tocenter, underscore.
- Added bibunits.
- Tested to work as-is with babelbib, bodegraph, fast-diagram, nicematrix, structmech.

v0.55: Various fixes.

misc fixes
- Fix: Extraneous space in file links, which also prevented *Calibre* EPUB conversions.
- Fix: Float optional argument regression.
- Fix: `\ForceHTMLTOC` with `\phantomsection`.
- Fix: Overfull boxes in `lateximages`.
- Fix: QED symbols in `lateximage`.

packages
- koma-script: Fix: Figure with `\centering`, etc.
- Added clrdblpg.

v0.54: Float `\centering`, improved image checks.

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **`pdflatex filename.tex`** or similar, after which *lwarpmk* may then be used with the new configuration files.

lwarpmk
- `lwarpmk limages` checks for the presence of the HTML version of the document and valid image references before attempting to create the `lateximages`.

- *lwarpmk*: Improved error message if configuration file does not exist.
- Added documentation for avoiding error with BibTeX and `\etalchar`. See section 8.6.9.

BibTeX

- Added documentation regarding polyglossia. See section 8.15.4.

polyglossia

- Added documentation regarding the use of macros in section names. See section 8.1.

macros in section names

document encoding

- Renamed and added package options:

⚠ New and revised encoding options

Old Package Option	New Package Option
`xdyFilename`	`xindyStyle`
`IndexLanguage`	`xindyLanguage`
–	`xindyCodepage`
–	`pdftotextEnc`

  Use these options along with inputenc or inputenx to process documents in an encoding other than UTF-8. See section 7.4.

floats with `\centering`, etc.

- Floats now honor `\centering`, `\raggedright`, `\raggedleft`, and their ragged2e equivalents, when placed directly after:

```
\begin{floattype}
\centering
```

misc. fixes

- tikz: `\pgfpicture`, fit, align, font.
- ragged2e: `\centering` etc.
- hyperref: `\hypertarget` was creating duplicate of `\label`.
- hyperref: Active chars inside `\hyperref`, `\hyperlink`.
- hyperref: `\ref` inside `\hyperlink` caused a nested HTML link.
- glossaries: Fix when not using babel or polyglossia.
- textcomp: `\textperthousand`.
- LaTeX core `verse` environment: line spacing.

⚠

- Removed `\citetitle`, adjusted `\attribution`.

packages

- memoir: Minor update for v3.7g.
- Added inputenx, bibunits, chngpage, forest, magaz, gridset.
- Prevents loading ae, aecc, t1enc, and wasysym.

v0.53: Improved image checks.

lwarpmk

- *lwarpmk*: Added a warning about corrupted images due to the need to recompile the document one more time.
- *lwarpmk*: Added the **`lwarpmk cleanlimages`** command.
- Added documentation for **`lwarpmk cleanlimages`** and **`lwarpmk pdftohtml`**.

v0.52: Improved footnotes, SVG math.

documentation

- Improved install instructions regarding `lwarp_baseline_marker.png`.
- Added documentation regarding footnotes in section headings, and footnotes with `\VerbatimFootnotes` from fancybox, fancyvrb. See section 8.5.4.
- Added documentation regarding font selection when using XeLaTeX or LuaLaTeX with fontspec and traditional font packages. See section 7.4.

SVG math

- Fix: Limit the number of background tasks when generating `lateximages`.

- Added user-adjustable SVG math font scaling. See section 84.3.
- Added warnings if `lwarp_baseline_marker.png` is not present, or if graphicx or graphics is not loaded.
- Improved `\ensuremath` hashing expansion.
- Fix: `equation*` with `split`.
- `tabbing` now works inside a `lateximage`. Use for math in `tabbing`.

MathJax
- Fix: MathJax script was not executing in some conditions.
- Added `\CustomizeMathJax` to add custom functions. See section 8.7.

footnotes
- Fix: Footnote numbering when using `HTMLDebugComments`.
- Fix: Footnote paragraph tags.
- Fix: `FootnoteDepth` defaults to `\subsubsection`.

misc. fixes
- Fix: `\kill` in a `lateximage`.
- Fix: `\FileDepth`, misc. others, when input encoding is not `utf8`.
- Fix: `\texorpdfstring` in a section name.

packages
- hyperref emulation: Fix for #, %, &, ~, _ characters in URLs.
- fancybox, fancyvrb: Initial support for `\VerbatimFootnotes`.
- nicefrac: Added with fix for `\ensuremath`.
- graphicx: Fix for option defaults. Added v1.1a/b options.
- endfloat: Updated for v2.6.
- url: Fixes for active characters.

2 Introduction

The lwarp project aims to allow a rich LaTeX document to be converted to a reasonable HTML5 interpretation, with only minor intervention on the user's part. No attempt has been made to force LaTeX to provide for every HTML-related possibility, and HTML cannot exactly render every possible LaTeX concept. Where compromise is necessary, it is desirable to allow the print output to remain typographically rich, and compromise only in the HTML conversion.

Several "modern" features of HTML5, CSS3, and SVG are employed to allow a fairly feature-rich document without relying on the use of JavaScript. Limited testing on older browsers shows that these new features degrade gracefully.

---

lwarp is a native LaTeX package, and operates by either patching or emulating various functions. Source-level compatibility is a major goal, but occasional user intervention is required in certain cases.

As a package running directly in LaTeX, lwarp has some advantages over other methods of HTML conversion. TeX itself is still used, allowing a wider range of TeX trickery to be understood. Lua expressions are still available with LuaTeX. Entire categories of LaTeX packages work as-is when used with lwarp: definitions, file handling, utilities, internal data structures and calculations, specialized math-mode typesetting for various fields of science and engineering, and anything generating plain-text output. Blocks of PDF output may be automatically converted to SVG images while using the same font and spacing as the original print document, directly supporting Ti*k*Z and `picture`. Numerous packages are easily adapted for HTML versions, either by loading and patching the originals, or by creating nullified or emulated replacements, and all without resorting to external programming. As a result, several hundred packages have already been adapted (table 2), and an uncounted number more work as-is.

Packages have been selected according to several criteria: perceived importance, popularity lists, recent CTAN updates, CTAN topics, mention in other packages, support by other HTML conversion methods, and from sample documents taken from public archives. These include some "obsolete" packages as well.[1]

Assistance is also provided for modifying the HTML output to suit the creation of EPUB documents, and for modifying the HTML output to ease import into a word processor.

---

pdflatex, *xelatex*, or *lualatex* may be used, allowing lwarp to process the usual image formats. While generating HTML output, SVG files are used in place of PDF. Other formats such as PNG and JPG are used as-is.

[1] An amazing number of decades-old packages are still in use today.

SVG images may be used for math, and are also used for `picture`, Ti*k*Z, and similar environments. The SVG format has better browser and e-book support than MathML (as of this writing), while still allowing for high-quality display and printing of images (again, subject to potentially bug-ridden[2] browser support).

Furthermore, SVG images allow math to be presented with the same precise formatting as in the print version. Math is accompanied by `<alt>` tags holding the LaTeX source for the expression, allowing it to be copy/pasted into other documents.[3] Custom LaTeX macros may be used as-is in math expressions, since the math is evaluated entirely inside LaTeX. An MD5 hash is used to combine multiple instances of the same inline math expression into a single image file, which then needs to be converted to SVG only a single time.

The MathJax JavaScript display engine may be selected for math display instead of using SVG images. Subject to browser support and Internet access, MathJax allows an HTML page to display math without relying on a large number of external image files.[4] lwarp maintains LaTeX control for cross-referencing and equation numbering, and attempts to force MathJax to tag equations accordingly.

A *texlua* program called *lwarpmk* is used to process either the print or HTML version of the document. A few external utility programs are used to finish the conversion from a LaTeX-generated PDF file which happens to have HTML5 tags, to a number of HTML5 plain-text files and accompanying images.

lwarp automatically generates the extra files necessary for the HTML conversion, such as CSS and `.xdy` files, and configuration files for the utility *lwarpmk*. Also included is a parallel version of the user's source document, `<sourcename>-html.tex`, which selects HTML output and then inputs the user's own source. This process allows both the printed and HTML versions to co-exist side-by-side, each with their own auxiliary files.

When requesting packages during HTML conversion, lwarp first looks to see if it has its own modified version to use instead of the standard LaTeX version. These `lwarp-packagename.sty` files contain code used to emulate or replace functions for HTML output.

---

[2] Firefox has had an on-again/off-again bug for quite some time regareding printing SVGs at high resolution.

[3] There seems to be some debate as to whether MathML is actually an improvement over LaTeX for sharing math. The author has no particular opinion on the matter, except to say that in this case LaTeX is much easier to implement!

[4] One SVG image file per math expression, except that duplicate inline math expressions are combined into a single file according to the MD5 hash function of its contents. A common scientific paper can easily include several thousand files, and in one case the MD5 hash cut the number of files in half and the rendering time by 30%.

2.1 Typesetting conventions

Font weight, family, and style are used to indicate various objects:

Table 1: Typesetting conventions

package	LaTeX package.
program	Program's executable name.
`option`	Program or package option.
`filename`	File name in the operating system.
Brand Name	Proper name for a program, operating system, etc.
`commands`	Commands to be entered by the user.
`code`	Program code.
`\macroname`	LaTeX macro.
`environment`	LaTeX environment.
`counter`	LaTeX counter.
`boolean`	LaTeX boolean.
`<element>`	HTML element.
`attribute`	HTML attribute.
User Interface	A user-interface item.
ACRO	Acronym.

subjects Blue-colored tags in the left margin aid in quickly identifying the subject of each paragraph. These are often the targets of index entries.

Prog Lwarp

index entries Black-colored tags in the left marign are used to identify programming objects such as files, packages, environments, booleans, and counters. Items without a tag are command macros. Each of these also appears in the index as individual entries, and are also listed together under "files", "packages", "environments", "booleans", and "counters".

⚠ warnings Special warnings are marked with a warning icon.

2.2 Supported packages and features

Table 2 lists some of the various LaTeX features and packages which may be used.

Package names are colored according to their support level:

name: Supported as-is.

name: Modifed to work with HTML output, and perhaps also as print output in SVG math or `latexmage` environments.

name: Emulated for HTML output.

name: Ignored for HTML output, but provides source-level compatibility.

MJ: Supported as-is for MathJax, subject to limitations.

MJ: Emulated for MathJax using custom macros, subject to limitations.

MJ: Ignored by MathJax, but may be used in the document source. May be converted to SVG images.

Table 2: LaTeX lwarp package — Supported features

Category	Status and supported features.
Engines:	DVI LaTeX, PDF LaTeX, XeLaTeX, LuaLaTeX, upLaTeX
LaTeX compiling:	*latexmk*, *make*, etc.
External compiling:	*perltex*, *pythontex*, sympytex
Classes:	article, book, report, scrartcl, scrbook, scrreprt, memoir, CJK-related as listed below.
Koma-script:	scrextend, scrhack, scrlayer. Others as listed below.
Memoir:	memhfixc
Beamer:	beamerarticle, but not the beamer class.
Languages:	babel, cjkpunct, impnattypo, luavlna, polyglossia, xeCJK, xevlna.
Chinese:	CTeX, ctex, upzhkinsoku, xpinyin, zhlineskip, zhspacing.
Japanese:	upLaTeX, LuaTeX-ja, gentombow, lltjext, plarray, plarydshln, plautopatch, plext, plextarray, plextarydshln, plextcolortbl, plextdelarray, pxatbegshi, pxeveryshi, pxftnright, pxgentombow, pxjahyper, pxpdfpages, pxpgfrcs, pxpgfmark, tascmac, zxjatype. bxjsarticle and related, ltjsarticle and related, luatexja, luatexja-fontspec, ujarticle and related, utarticle and related.
Korean:	kotex, luatexko, xetexko.

Page layout:	2in1, 2up, a4, a4wide, a5comb, addlines, anysize, atbegshi, balance, blowup, booklet, bophook, bounddvi, bxpapersize, canoniclayout, centerlastline, changelayout, changepage, chngpage, clrdblpg, continue, draftcopy, draftfigure, draftwatermark, ebook, everyshi, fancyhdr, fancytabs, flippdf, fullminipage, fullpage, fwlw, geometry, gmeometric, grid, grid-system, gridset, layaureo, layout, layouts, leading, lscape, ltxgrid, nccfancyhdr, notespages, nowidow, pagegrid, pagesel, parallel, parcolumns, pbalance, pdfcolparallel, pdfcolparcolumns, pdfcrypt, pdflscape, pdfprivacy, preview, ragged2e, returntogrid, rmpage, scrlayer-scrpage, scrpage2, setspace, selectp, textarea, threadcol, thumb, thumbs, titleps, tocenter, turnthepage, twoup, typearea, underlin, vmargin, watermark, widows-and-orphans, zwpagelayout.
Sectioning:	Adds `FileDepth` for splitting the HTML output. Files may be numbered sequentially or named according to section name. Common short words and punctuation are removed from the filenames. anonchap, bsheaders, decorule, fncychap, froufrou, hypbmsec, indentfirst, quotchap, section, sectionbreak, secdot, sectsty, titlesec, tocvsec2.
Table of contents, figures, tables:	Supported, with hyperlinks. etoc, minitoc, multitoc, shorttoc, tableof, titletoc, tocbasic, tocbibind, tocdata, tocloft, tocstyle, tocvsec2.
Title page:	`\maketitle`, `titlepage`, authblk, authoraftertitle, titling.
Front & back matter:	abstract, appendix.
Indexing:	*makeindex*, *xindy*, and *xindex* are supported, with hyperlinks. gindex, hvindex, idxlayout, imakeidx, index, makeidx, repeatindex, splitidx, varindex, xindex.
Glossary:	gloss, glossaries and *xindy*, nomencl.
Bibliography:	babelbib, bibtopic, backref, biblatex, bibunits, chapterbib, cite, citeref, collref, drftcite, hypernat, jurabib, mcite, mciteplus, multibib, natbib, notes2bib, splitbib, showtags.
Cross-references:	bookmark, breakurl, cleveref, fancyref, hypdestopt, hyperref, perpage, prettyref, titleref, url, varioref, xcite, xr, xr-hyper, xurl, zref.
Margin notes:	marginal, marginfit, marginfix, scrlayer-notecolumn, versonotes.

Footnotes:	Adds `FootnoteDepth` to print footnotes at section breaks. MathJax emulation for `\footnote`, and also as marked in the following: bigfoot, dblfnote, endheads, endnotes[MJ], enotez[MJ], fixfoot, fnbreak, fnpara, fnpct, fnpos, footmisc, footnote, footnotebackref, footnoterange, footnpag, manyfoot, marginnote[MJ], nccfoots[MJ], pagenote[MJ], parnotes[MJ], pdfcolfoot, pfnote, sepfootnotes, sidenotes[MJ], tablefootnote.
Math:	Converted to SVG images with HTML `<alt>` tags containing the LaTeX source for the math expression. MathJax supported as an alternative. amsmath[MJ]: $\mathcal{AMS}$ environments are supported. User-defined macros are available during conversion, due to native LaTeX processing.
Theorems:	Native LaTeX theorems, amsthm, apxproof, ntheorem, shadethm, theorem, thmbox, thmtools.
Additional math:	Math fonts via SVG images, accents[MJ], amscd[MJ], amscdx, autobreak[MJ], autonum, backnaur[MJ], bm[MJ], braket[MJ], breqn[MJ], bussproofs[MJ], cases[MJ], centernot[MJ], cmbright[MJ], colonequals[MJ], decimal[MJ], delarray, DotArrow[MJ], dotlessi[MJ], dotlessj[MJ], esvect[MJ], extarrows[MJ], fixmath[MJ], fouridx[MJ], fourier[MJ], guass, hhtensor[MJ], icomma[MJ], isomath[MJ], jkmath, kpfonts[MJ], kpfonts-otf[MJ], leftidx[MJ], libertinust1math[MJ], mathalpha[MJ], mathastext[MJ], mathcomp[MJ], mathdesign[MJ], mathdots[MJ], mathfixs[MJ], mathpazo[MJ], mathptmx[MJ], mathpunctspace[MJ], mathspec[MJ], mathtools[MJ], mattens[MJ], maybemath[MJ], mdwmath[MJ], mismath[MJ], mleftright[MJ], multiobjective[MJ], nccmath[MJ], nicematrix[MJ], noitcrul[MJ], newpxmath[MJ], newtxmath[MJ], newtxsf[MJ], pb-diagram, pxfonts[MJ], resizegather[MJ], rmathbr[MJ], scalerel[MJ], shuffle[MJ], skmath[MJ], stackrel[MJ], statex2[MJ], statistics, statmath[MJ], subsupscripts[MJ], tensind, tensor[MJ], textualicomma[MJ], txfonts[MJ], txgreeks[MJ], unicode-math[MJ], upgreek[MJ], ushort[MJ], witharrows[MJ], xfakebold[MJ], xy. Many others work as-is.
Display math with `\displaymathother`:	Complicated math objects in display math, such as tikz-cd, etc.
Units and fractions:	nicefrac[MJ], SIunits[MJ], siunitx[MJ], units[MJ], unitsdef, xfrac[MJ].

Floats:	Appear where declared. capt-of, caption, cutwin, dblfloatfix, endfloat, fewerfloatpages, fix2col, flafter, float, floatflt, floatrow, fltrace, ftcap, hypcap, keyfloat, morefloats, multicap, newfloat, nonfloat, picinpar, placeins, rotfloat, stfloats, subcaption, subfig, subfigure, subfloat, swfigure, topcapt, trivfloat, wrapfig, wrapfig2.
Tabular:	`tabular` environment, array[MJ], arydshln[MJ], bigdelim[MJ], bigstrut[MJ], booktabs[MJ], colortbl[MJ], ctable, dcolumn, diagbox, hhline[MJ], longtable, ltablex, ltxtable, multirow[MJ], supertabular, tabularx, tabulary, threeparttable, threeparttablex, widetable, xltabular, xtab.
Graphics:	graphics and graphicx. `\includegraphics` supports `width`, `height`, `origin`, `angle`, and `scale` tags, and adds `class`. References to PDF files are changed to SVG, other image types are accepted as well. `\rotatebox` and `\scalebox` are supported as well as HTML can handle. rotating is emulated but all objects are unrotated in HTML. `picture`, tikz, and xy are converted to an SVG image. asymptote, curves, datatool, eepic, epsf, epsfig, epstopdf, figsize, fitbox, grffile, lpic, luamplib, media9, movie15, multimedia, overpic, pict2e, pinlabel, psfrag, psfragx, pst-eps, pstool, pstricks, rlepsf, rviewport, svg, svg-extract, tikz, tikz-3dplot, tikz-imagelabels, xy
xcolor:	Full package color names, any color models, and mixing. `\textcolor`, `\colorbox`, `\fcolorbox`. Enhanced for HTML compatibility.
Lists:	Standard LaTeX environments, enumerate, enumitem, eqlist, hang, listliketab, paralist.
Environments:	Standard LaTeX environments.
Paragraphs, `minipage`, `\parbox`:	Some HTML5-imposed limitations. Nested minipages are supported. eqparbox, fancypar, minibox, pbox, shapepar.
Quotations:	copyrightbox, csquotes, epigraph, quoting, verse.
Verbatim:	fancyvrb, fvextra, moreverb, shortvrb, verbatim.
Frames:	boxedminipage, boxedminipage2e, fancybox, fbox[MJ], framed, mdframed, niceframe, shadow, tcolorbox[MJ], vertbars.
Multi-columns:	adjmulticol, multicol, multicolrule, vwcol.
Margins:	fullwidth, hanging, midpage.
Line numbering:	fnlineno, lineno.

Direct formatting:	`\emph`, `\textsuperscript`, `\textbf`, etc are supported. `\bfseries`, etc. are only supported in some cases. cancel[MJ], ellipsis, embrac, enparen, hyphenat, lettrine, lips, lua-check-hyphen, luacolor, magaz, moresize, nolbreaks, normalcolor, pdfcol, pdfcolmk, pdfrender, realscripts, relsize[MJ], scalefnt, seqsplit[MJ], soul, soulpos, soulutf8, stackengine, textfit, thinsp, trimclip, truncate, ulem, umoline, underscore, uspace, xellipsis.
Acronyms:	acro, acronym.
Ordinals:	engord, fmtcount, nth.
Text ligatures:	Ligatures for symbols are supported. Ligatures for f, q, t are intentionally turned off because many simpler browsers do not display them correctly. Modern full-featured browsers re-create these ligatures on-the-fly.
Horizontal space:	HTML output for thin-unbreakable, unbreakable, `\enskip`, `\quad`, `\qquad`, `\hspace`.
Rules:	`\rule` with width, height, raise, text color.
HTML reserved characters:	`\&`, `\textless`, and `\textgreater` are converted to HTML entities.
Fonts:	Used as-is. Appear in SVG math expressions or embedded image environments. fontaxes, nfssext-cfr, slantsc, tabfigures. Tested to work as-is: Special font macros in cfr-lm and others which use nfssext-cfr. Also see the math section for math and MathJax support for math font packages.
Symbols:	Native LaTeX diacriticals, academicons, amssymb[MJ], bbding, ccicons, chemgreek, dingbat, euro, eurosym, fontawesome, fontawesome5, gensymb[MJ], latexsym[MJ], marvosym, metalogo, metalogox, pifont, textalpha, textcomp[MJ], textgreek, typicons, xunicode.
Files:	attachfile, attachfile2, hyperxmp, inputtrc, intopdf, pdfpages, pdfx, xmpincl.

Science and engineering:	algorithm2e, algorithmicx, ar[MJ], askmaps, axodraw2, bitpattern, blochsphere, bodegraph, bohr, bytefield, chemfig, chemformula, chemgreek, chemmacros, chemnum, circuitikz, econometrics[MJ], elements, engtlc[MJ], fast-diagram, ghsystem, hepnicenames, heppennames, hepunits[MJ], isotope[MJ], karnaughmap, karnaugh-map, keystroke, listings, listingsutf8, linop, menukeys, mhchem[MJ], minted, pgfgantt, phfqit, physics[MJ], physunits[MJ], plimsoll[MJ], qcircuit, register, simplebnf, simpler-wick, slashed[MJ], steinmetz[MJ], structmech, struktex, syntaxdi, tikz-karnaugh, tikzcodeblocks, venndiagram
Arts and humanities:	foreign, forest, lyluatex, musicography, nameauth, octave, phonrule, piano, schemata, semantic-markup, tikz-dependency, vowel, xpiano
Academic:	academicons, classicthesis, doi, doipubmed, orcidlink[MJ], termcal
Admonitions:	awesomebox, notes.
Editorial:	changebar, changelog, changes, easy-todo, easyReview, ed, errata, fixme, fixmetodonotes, pdfcomment[MJ], pdfmarginpar, todo, todonotes, tram, xechangebar.
Accessibility:	accessibility[MJ], accsupp[MJ], axessibility[MJ], pdfcomment[MJ], repltext[MJ], tagpdf.
Package handling:	catoptions.
Debug:	chkfloat, cmdtrack, dprogress, lipsum, lua-visual-debug, mwe, refcheck, showlabels, showkeys, srcltx, srctex, vpe, xbmks.
Working as-is:	Various utility, calculation, file, and text-only packages, such as calc, fileerr, somedefs, trace, xspace. Also, most math-only packages, including specialized typesetting for various fields of science and engineering.

3 Alternatives

Summarized below are several other ways to convert a LaTeX or other document to HTML. Where an existing LaTeX document is to be converted to HTML, lwarp may be a good choice. For new projects with a large number of documents, it may be worth investigating the alternatives before decided which path to take.

3.1 internet class

`internet` (*Cls*) The closest to lwarp in design principle is the `internet` class by Andrew Stacey — an interesting project which directly produces several versions of markdown, and also HTML and EPUB. https://github.com/loopspace/latex-to-internet

3.2 TeX4ht

`TeX4ht` (*Prog*) http://tug.org/tex4ht/

`htlatex` (*Prog*)

This system uses native LaTeX processing to produce a DVI file containing special commands, and then uses additional post-processing for the HTML conversion by way of numerous configuration files. In some cases lwarp provides a better HTML conversion, and it supports a different set of packages. TeX4ht produces several other forms of output beyond HTML, including ODT and a direct path to EPUB, and is still being developed.

3.3 Translators

These systems use external programs to translate a subset of LaTeX syntax into HTML. Search for each on CTAN (http:\ctan.org).

`Hevea` (*Prog*) **HeVeA:** http://hevea.inria.fr/ (not on CTAN)

`TtH` (*Prog*) **TtH:** http://hutchinson.belmont.ma.us/tth/

`GELLMU` (*Prog*) **GELLMU:** http://www.albany.edu/~hammond/gellmu/

`LaTeXML` (*Prog*) **LaTeXML:** http://dlmf.nist.gov/LaTeXML/

`Plastex` (*Prog*) **PlasTeX:** https://github.com/tiarno/plastex

`LaTeX2HTML` (*Prog*) **LaTeX2HTML:** http://www.latex2html.org/
and http://ctan.org/pkg/latex2html.

`TeX2page` (*Prog*) **TeX2page:** http://ds26gte.github.io/tex2page/index.html

Finally, GladTeX may used to directly insert LaTeX math into HTML:

`GladTeX` (*Prog*) **GladTeX:** http://humenda.github.io/GladTeX/

3.4 AsciiDoc and Asciidoctor

AsciiDoc is one of the most capable markup languages, providing enough features to produce the typical technical-writing document with cross-references, and it writes LaTeX and HTML.

AsciiDoc (*Prog*) **Asciidoctor:** http://asciidoctor.org/ (More active.)

AsciiDoctor (*Prog*) **AsciiDoc:** http://asciidoc.org/ (The original project.)

3.4.1 Asciidoctor-LaTeX

The Asciidoctor-LaTeX project is developing additional LaTeX-related features.

Asciidoctor-LateX:
http://www.noteshare.io/book/asciidoctor-latex-manual
Asciidoctor-LaTeX (*Prog*) https://github.com/asciidoctor/asciidoctor-latex

3.5 Pandoc

Pandoc (*Prog*) A markup system which also reads and writes LaTeX and HTML.

Pandoc: http://pandoc.org/

(Watch for improvements in cross-references to figures and tables.)

3.6 Word processors

Word (*Prog*)
LibreOffice (*Prog*)
OpenOffice (*Prog*)
It should be noted that the popular word processors have advanced through the years in their abilities to represent math with a LaTeX-ish input syntax, unicode math fonts, and high-quality output, and also generate HTML with varying success. See recent developments in Microsoft ® *Word* ® and LibreOffice ™ *Writer*.

3.7 Commercial systems

Adobe (*Prog*)
FrameMaker (*Prog*)
InDesign (*Prog*)
Flare (*Prog*)
Madcap (*Prog*)
Likewise, several professional systems exist whose abilities have been advancing in the areas of typesetting, cross-referencing, and HTML generation. See Adobe ® *FrameMaker* ®, Adobe *InDesign* ®, and Madcap *Flare* ™.

3.8 Comparisons

AsciiDoc, Pandoc, and various other markup languages typically have a syntax which tries to be natural and human-readable, but the use of advanced features tends to require many combinations of special characters, resulting in a complicated mess of syntax. By contrast, LaTeX spells things out in readable words but takes longer to type, although integrated editors exist which can provide faster

entry and a graphic user interface. For those functions which are covered by the typical markup language it is arguable that LaTeX is comparably easy to learn, while LaTeX provides many more advanced features where needed, along with a large number of pre-existing packages which provide solutions to numerous common tasks.

Text-based document-markup systems share some of the advantages of LaTeX vs. a typical word processor. Documents formats are stable. The documents themselves are portable, work well with revision control, do not crash or become corrupted, and are easily generated under program control. Formatting commands are visible, cross-referencing is automatic, and editing is responsive. Search/replace with regular expressions provides a powerful tool for the manipulation of both document contents and structure. Markup systems and some commercial systems allow printed output through a LaTeX back end, yielding high-quality results especially when the LaTeX template is adjusted, but they lose the ability to use LaTeX macros and other LaTeX source-document features.

The effort required to customize the output of each markup system varies. For print output, LaTeX configuration files are usually used. For HTML output, a CSS file will be available, but additional configuration may require editing some form of control file with a different syntax, such as XML. In the case of lwarp, CSS is used, and much HTML output is adjusted through the usual LaTeX optional macro parameters, but further customization may require patching LaTeX code.

The popular word processors and professional document systems each has a large base of after-market support including pre-designed styles and templates, and often include content-management systems for topic reuse.

4 Installation

Table 3 shows the tools which are used for the LaTeX to HTML conversion. In most cases, these will be available via the standard package-installation tools.

Detailed installation instructions follow.

Table 3: Required software programs

Provided by your LaTeX distribution:

From TeXLive: http://tug.org/texlive/.

LaTeX: *pdflatex*, *xelatex*, or *lualatex*.

The lwarp package: This package.

The *lwarpmk* utility: Provided along with this package. This should be an operating-system executable in the same way that *pdflatex* or *latexmk* is. It is possible to have the lwarp package generate a local copy of *lwarpmk* called `lwarpmk.lua`. See table 4.

***luatex*:** Used by the *lwarpmk* program to simplify and automate document generation.

***xindy*:** The *xindy* program is used by lwarp to create indexes. On a MiKTeX system this may have to be acquired separately, but it is part of the regular installer as of mid 2015.

***latexmk*:** Optionally used by *lwarpmk* to compile LaTeX code. On a MiKTeX system, *Perl* may need to be installed first.

***pdfcrop*:** Used to pull images out of the LaTeX PDF.

Poppler PDF utilities:

***pdftotext*:** Used to convert PDF to text.

***pdfseparate*:** Used to pull images out of the LaTeX PDF.

***pdftocairo*:** Used to convert images to SVG.

These might be provided by your operating-system package manager, and MiKTeX provides miktex-poppler-bin-* packages.

From Poppler: poppler.freedesktop.org.

For MacOS®, see https://brew.sh/, install *Homebrew*, then

Enter ⇒ **`brew install poppler`**

For Windows, see MikTeX miktex-poppler-bin-*, or:
https://sourceforge.net/projects/poppler-win32/ and:
http://blog.alivate.com.au/poppler-windows/

***Perl*:**

This may be provided by your operating-system package manager, and may be required for some of the Poppler PDF utilities.

strawberryperl.com (recommended), perl.org

Automatically downloaded from the internet as required:

MathJax: Optionally used to display math. From: mathjax.org

4.1 Installing the lwarp package

There are several ways to install lwarp. These are listed here with the preferred methods listed first:

Pre-installed: Try entering into a command line:

Enter ⇒ **`kpsewhich lwarp.sty`**

If a path to `lwarp.sty` is shown, then lwarp is already installed and you may skip to the next section.

TEX Live: If using a TEX Live distribution, try installing via *tlmgr*:

Enter ⇒ **`tlmgr install lwarp`**

MiKTEX:

1. For newer versions of MiKTEX, install or update lwarp using the *MiKTeX Console* program.
2. For older versions of MiKTEX, to install lwarp the first time, use the *MiKTeX Package Manager (Admin)*. To update lwarp, use *MiKTeX Update (Admin)*.
3. Either way, also update the package miktex-misc, which will install and update the *lwarpmk* executable.

Operating-system package: The operating-system package manager may already have lwarp, perhaps as part of a set of TEX-related packages.

CTAN TDS archive: lwarp may be downloaded from the Comprehensive TEX Archive:

1. See http://ctan.org/pkg/lwarp for the lwarp package.
2. Download the TDS archive: `lwarp.tds.zip`
3. Find the TEX `local` directory:

   TEX Live:

   Enter ⇒ **`kpsewhich -var-value TEXMFLOCAL`**

   MiKTEX:

   In the **Settings** window, **Roots** tab, look for a local TDS root.

   This should be something like:

   `/usr/local/texlive/texmf-local/`
4. Unpack the archive in the TDS local directory.
5. Renew the cache:

   Enter ⇒ **`mktexlsr`**

   — or —

   Enter ⇒ **`texhash`**

   Or, for WINDOWS MiKTEX, start the program called *MiKTeX Settings (Admin)* and click on the button called **Refresh FNDB**.

CTAN `.dtx` and `.ins` files: Another form of TEX package is `.dtx` and `.ins` source files. These files are used to create the documentation and `.sty` files.

1. See http://ctan.org/pkg/lwarp for the lwarp package.
2. Download the zip archive `lwarp.zip` into your own `lwarp` directory.
3. Unpack `lwarp.zip`.

4. Locate the contents `lwarp.dtx` and `lwarp.ins`
5. Create the `.sty` files:

   Enter ⇒ **`pdflatex lwarp.ins`**
6. Create the documentation:

```
pdflatex lwarp.dtx  (several times)
makeindex -s gglo.ist -o lwarp.gls lwarp.glo
makeindex -s gind.ist lwarp.idx
pdflatex lwarp.dtx  (several times)
```

7. Copy the `.sty` files somewhere such as the TEX Live local tree found in the previous CTAN TDS section, under the subdirectory:

   `<texlocal>/tex/latex/local/lwarp`
8. Copy `lwarp_baseline_marker.png` and `lwarp_baseline_marker.eps` to the same place as the `.sty` files.
9. Copy the documentation `lwarp.pdf` to a `source` directory in the local tree, such as:

   `<texlocal>/doc/local/lwarp`
10. Renew the cache:

    Enter ⇒ **`mktexlsr`**

    — or —

    Enter ⇒ **`texhash`**

    Or, for WINDOWS MiKTEX, start the program called *MiKTeX Settings (Admin)* and click on the button called **Refresh FNDB**.
11. See section 4.2.1 to generate your local copy of *lwarpmk*.
12. Once the local version of `lwarpmk.lua` is installed, it may be made available system-wide as per section 4.2.

Project-local CTAN `.dtx` and `.ins` files: The `.dtx` and `.ins` files may be downloaded to a project directory, then compiled right there, alongside the document source files. The resultant `*.sty` and `lwarpmk.lua` files may be used as-is, so long as they are in the same directory as the document source. The files `lwarp_baseline_marker.png` and `lwarp_baseline_marker.eps` must also be copied as well. This approach is especially useful if you would like to temporarily test lwarp before deciding whether to permanently install it.

Just testing!

4.2 Installing the *lwarpmk* utility

(Note: If *lwarpmk* is not already installed, it is easiest to use a local copy instead of installing it system-wide. See section 4.2.1.)

After the lwarp package is installed, you may need to setup the *lwarpmk* utility:

1. At a command line, try executing **`lwarpmk`**. If the *lwarpmk* help message appears, then *lwarpmk* is already set up. If not, it is easiest to generate and use a local copy. See section 4.2.1.
2. For MiKTEX, try updating the miktex-misc package. This may install the *lwarpmk* executable for you.

   Otherwise, continue with the following:

3. Locate the file `lwarpmk.lua`, which should be in the `scripts` directory of the TDS tree. On a TEX Live or MiKTEX system you may use

   Enter ⇒ **`kpsewhich lwarpmk.lua`**

   (If the file is not found, you may also generate a local copy and use it instead. See section 4.2.1.)

4. Create *lwarpmk*:

   Unix: Create a symbolic link and make it executable:

   (a) Locate the TEX Live binaries:
   Enter ⇒ **`kpsewhich -var-value TEXMFROOT`**
   This will be something like:
   `/usr/local/texlive/<year>`
   The binaries are then located in the `bin/<arch>` directory under the root:
   `/usr/local/texlive/<year>/bin/<architecture>/`
   In this directory you will find programs such as *pdflatex* and *makeindex*.

   (b) In the binaries directory, create a new symbolic link from the binaries directory to `lwarpmk.lua`:
   Enter ⇒ **`ln -s <pathtolwarpmk.lua> lwarpmk`**

   (c) Make the link executable:
   Enter ⇒ **`chmod 0755 lwarpmk`**

   WINDOWS TEX Live: Create a new `lwarpmk.exe` file:

   (a) Locate the TEX Live binaries as shown above for UNIX.

   (b) In the binaries directory, make a *copy* of `runscript.exe` and call it `lwarpmk.exe` This will call the copy of `lwarpmk.lua` which is in the `scripts` directory of the distribution.

   WINDOWS MiKTEX: Create a new `lwarpmk.bat` file:

   (a) Locate the MiKTEX binaries. These will be in a directory such as:
   `C:\Program Files\MiKTeX 2.9\miktex\bin\x64`
   In this directory you will find programs such as `pdflatex.exe` and `makeindex.exe`.

   (b) Create a new file named `lwarpmk.bat` containing:
   `texlua "C:\Program Files\MiKTeX 2.9\scripts\lwarp\lwarp.texlua" %*`
   This will call the copy of `lwarpmk.lua` which is in the `scripts` directory of the distribution.

4.2.1 Using a local copy of *lwarpmk*

It is also possible to use a local version of *lwarpmk*:

1. When compiling the tutorial in section 5, use the `lwarpmk` option for the lwarp package:

   ```
   \usepackage[lwarpmk]{lwarp}
   ```

2. When the tutorial is compiled with *pdflatex*, the file `lwarpmk.lua` will be generated along with the other configuration files.

3. `lwarpmk.lua` may be used for this project:

Unix:

(a) Make `lwarpmk.lua` executable:

Enter ⇒ **`chmod 0755 lwarpmk.lua`**

(b) Compile documents with

Enter ⇒ **`./lwarpmk.lua html`**

Enter ⇒ **`./lwarpmk.lua print`**

etc.

(c) It may be useful to rename or link to a version without the `.lua` suffix.

WINDOWS:

Compile documents with either of the following, depending on which command shell is being used:

Enter ⇒ **`texlua lwarpmk.lua html`**

Enter ⇒ **`texlua lwarpmk.lua print`**

etc.

Or:

Enter ⇒ **`lwarpmk html`**

Enter ⇒ **`lwarpmk print`**

etc.

4.3 Installing additional utilities

To test for the existence of the additional utilities:

Enter the following in a command line. If each programs' version is displayed, then that utility is already installed. See table 3 on page 77.

Enter ⇒ **`luatex --version`**

Enter ⇒ **`xindy --version`**

Enter ⇒ **`latexmk --version`**

Enter ⇒ **`perl --version`**

Enter ⇒ **`pdfcrop --version`**

Enter ⇒ **`pdftotext -v`**

Enter ⇒ **`pdfseparate --version`**

Enter ⇒ **`pdftocairo -v`**

To install *xindy*, *latexmk*, and *pdfcrop*:

The TEX utilities *xindy*, *latexmk*, and *pdfcrop* may be installed in *TexLive* with *tlmgr*, installed by *MiKTeX*, provided by your operating system's package manager, or downloaded from the *CTAN* archive:

http://ctan.org/pkg/xindy
http://ctan.org/pkg/latexmk
http://ctan.org/pkg/pdfcrop

`pdftotext` (*Prog*) [requirement]
`pdfseparate` (*Prog*) [requirement]
`pdftocairo` (*Prog*) [requirement]

To install the POPPLER utilities to a UNIX/LINUX system:

The tools from the POPPLER project should be provided by your operating system's package manager.

To install the POPPLER utilities to a MACOS machine:

1. Install *Homebrew* from https://brew.sh/:

Enter ⇒ **`/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`**

2. Install the POPPLER utilities:
   Enter ⇒ `brew install poppler`

To install the POPPLER utilities to a WINDOWS machine:

If using MikTEX, install a miktex-poppler-bin-* package. Otherwise:

1. See table 3 on page 77.
2. Download and extract the POPPLER utilities *pdftotext*, *pdfseparate*, and *pdfseparate* to a directory, such as `Poppler`.
3. In the **Start** window, type "Path" to search for results related to Path. Or, open the control panel and search for "Path".
4. Choose **Edit the system environment variables** in the control panel.
5. Choose the **Environment Variables** button.
6. Choose the **Path** variable, then the **Edit** button.
7. Choose the **New** button to make an additional entry.
8. Enter the `bin` directory of the POPPLER utilities, such as:
   `C:\Users\<myname>\Desktop\Poppler\poppler-0.5_x86\poppler-0.5\bin`
   Be sure to include `\bin`.
9. Click **Ok** when done.

`perl` (*Prog*) [requirement]

To install PERL to a WINDOWS machine:

1. Download and install a version of PERL, such as STRAWBERRY PERL, to a directory without a space in its name, such as `C:\Strawberry`.
2. Edit the **Path** as seen above for the POPPLER utilities.
3. Enter the `bin` directory of the *perl* utility, such as:
   `C:\Strawberry\perl\bin`
   Be sure to include `\bin`.
4. Click **Ok** when done.

Any utilities installed by hand must be added to the PATH.

5 Tutorial

This section shows an example of how to create an lwarp document.

Need help?

See the General Index for "how-to", and the Troubleshooting Index if something doesn't work. A Troubleshooting section is also available. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category.

5.1 Starting a new project

1. Create a new project directory called `tutorial`.

`tutorial.tex` (*file*)

2. Inside the `tutorial` directory, create a new file called `tutorial.tex`. This may be done several ways:

   Copy from the documentation PDF:

   A listing is in fig. 1, which may be copied/pasted from the figure directly into your own editor, depending on the quality of the PDF viewer and editor, or:

   Copy from the lwarp documentation directory:

   `lwarp_tutorial.txt` (*file*)

   Another copy may be found by entering into a command line:

   Enter ⇒ **`texdoc -l lwarp_tutorial.txt`**

   ⚠ Note: `.txt` suffix!

   This should be in the `doc/latex/lwarp/` directory along with this PDF documentation. Copy `lwarp_tutorial.txt` directly into your `tutorial` directory, renamed as `tutorial.tex`.

   ⚠ Bad formatting!

   When using Windows, use an editor other than Notepad, since Notepad does not accept the end-of-line from a Unix text file.

3. Compile the project:

   Enter ⇒ **`pdflatex tutorial.tex`**

   (several times)

   (*xelatex* or *lualatex* may be used as well. lwarp also supports DVI *latex* for use with `.eps` images.)

4. View the resulting `tutorial.pdf` with a PDF viewer.

A number of new files are created when `tutorial.tex` is compiled, as shown in table 4. These files are created by the `lwarp` package.

(Two of the new files are configuration files for the helper program *lwarpmk*. Whenever a print version of the document is created, the configuration files for *lwarpmk* are updated to record the operating system, LaTeX engine (*latex*, *pdflatex*, *xelatex*, or *lualatex*), the filenames of the source code and HTML output, and whether the additional helper program *latexmk* will be used to compile the document.)

Figure 1: tutorial.tex listing

Note: There are two pages!

```
% Save this as tutorial.tex for the lwarp package tutorial.

\documentclass{book}

\usepackage{iftex}

% --- LOAD FONT SELECTION AND ENCODING BEFORE LOADING LWARP ---

\ifPDFTeX
\usepackage{lmodern}            % pdflatex or dvi latex
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\else
\usepackage{fontspec}           % XeLaTeX or LuaLaTeX
\fi

% --- LWARP IS LOADED NEXT ---
\usepackage[
%   HomeHTMLFilename=index,     % Filename of the homepage.
%   HTMLFilename={node-},       % Filename prefix of other pages.
%   IndexLanguage=english,      % Language for xindy index, glossary.
%   latexmk,                    % Use latexmk to compile.
%   OSWindows,                  % Force Windows. (Usually automatic.)
%   mathjax,                    % Use MathJax to display math.
]{lwarp}
% \boolfalse{FileSectionNames}  % If false, numbers the files.

% --- LOAD PDFLATEX MATH FONTS HERE ---

% --- OTHER PACKAGES ARE LOADED AFTER LWARP ---
\usepackage{makeidx} \makeindex
\usepackage{xcolor}             % (Demonstration purposes only.)
\usepackage{hyperref,cleveref}  % LOAD THESE LAST!

% --- LATEX AND HTML CUSTOMIZATION ---
\title{The Lwarp Tutorial}
\author{Some Author}
\setcounter{tocdepth}{2}        % Include subsections in the \TOC.
\setcounter{secnumdepth}{2}     % Number down to subsections.
\setcounter{FileDepth}{1}       % Split \HTML\ files at sections
\booltrue{CombineHigherDepths}  % Combine parts/chapters/sections
\setcounter{SideTOCDepth}{1}    % Include subsections in the side\TOC
\HTMLTitle{Webpage Title}       % Overrides \title for the web page.
\HTMLAuthor{Some Author}        % Sets the HTML meta author tag.
\HTMLLanguage{en-US}            % Sets the HTML meta language.
\HTMLDescription{A description.}% Sets the HTML meta description.
\HTMLFirstPageTop{Name and \fbox{HOMEPAGE LOGO}}
\HTMLPageTop{\fbox{LOGO}}
\HTMLPageBottom{Contact Information and Copyright}
\CSSFilename{lwarp_sagebrush.css}

\begin{document}

\maketitle                      % Or titlepage/titlingpage environment.
```

```
% An article abstract would go here.

\tableofcontents                % MUST BE BEFORE THE FIRST SECTION BREAK!
\listoffigures

\chapter{First chapter}

\section{A section}

This is some text which is indexed.\index{Some text.}

\subsection{A subsection}

See \cref{fig:withtext}.

\begin{figure}\begin{center}
\fbox{\textcolor{blue!50!green}{Text in a figure.}}
\caption{A figure with text\label{fig:withtext}}
\end{center}\end{figure}

\section{Some math}

Inline math: $r = r_0 + vt - \frac{1}{2}at^2$
followed by display math:
\begin{equation}
a^2 + b^2 = c^2
\end{equation}


\begin{warpprint}   % For print output ...
\cleardoublepage    % ... a common method to place index entry into TOC.
\phantomsection
\addcontentsline{toc}{chapter}{\indexname}
\end{warpprint}
\ForceHTMLPage      % HTML index will be on its own page.
\ForceHTMLTOC       % HTML index will have its own toc entry.
\printindex

\end{document}
```

Table 4: Configuration files created by print version

tutorial.pdf: The PDF output from LaTeX. The print version of the document.

tutorial_html.tex: A small .tex file used to create a parallel HTML version of the document, which co-exists with usual the PDF version, and which will have its own auxiliary files. In this way, both PDF and HTML documents may co-exist side-by-side.

Auxiliary files: The usual LaTeX files .aux, .log, .out, .toc, .lof, .idx. When an HTML version of the document is created, _html versions of the auxiliary files will also be generated.

lwarpmk.conf: A configuration file for *lwarpmk*, which is used to automate the compilation of PDF or HTML versions of the document.

tutorial.lwarpmkconf: Another configuration file used by *lwarpmk*, which is only useful if you wish to have several projects residing in the same directory.

.css files: lwarp.css, lwarp_formal.css, lwarp_sagebrush.css These files are standard for lwarp, and are not meant to be modified by the user.

sample_project.css: An example of a user-customized CSS file, which may be used for project-specific changes to the lwarp defaults.

lwarp.ist: Used by lwarp while creating an index using *makeindex*. This file should not be modified by the user. A custom file may be used instead, if necessary.

lwarp.xdy: Used by lwarp while creating an index using *xindy*. This file should not be modified by the user. A custom file may be used instead, if necessary.

lwarp_one_limage.txt: For WINDOWS only. Used to process SVG images in the background. Copied to lwarp_one_limage.cmd when images are generated.

lwarp_mathjax.txt: Inserted into the HTML files when MATHJAX is used to display math. Do not modify, see \MathJaxFilename instead.

comment_*.cut: Temporary files used by lwarp to conditionally process blocks of text. These files may be ignored.

When the lwarpmk option is given to the lwarp package:

lwarpmk.lua: A local copy of the *lwarpmk* utility.

On UNIX-related operating systems this file must be made executable:

```
chmod u+x lwarpmk.lua
```

This may be useful to have to archive with a project for future use.

5.2 Compiling the print version with *lwarpmk*

The *lwarpmk* utility program is used to compile either the printed or the HTML version of the document.

`lwarpmk print` is used to recompile a printed version of the document.

⚠ Enable *lwarpmk*

1. If you have not yet done so, add `\usepackage{lwarp}` to the document, then compile the project a single time using *pdflatex*, *lualatex*, or *xelatex*. This generates the file `lwarpmk.conf`, which then allows the *lwarpmk* program to be used.

2. Re-compile the print version:

   Enter ⇒ **`lwarpmk print`**

   lwarpmk prints an introduction then checks to see if the document must be recompiled. If it seems that the files are up-to-date, then *lwarpmk* informs you of that fact and then exits.

3. Make a small change in the original document, such as adding a space character.

4. Recompile again.

   Enter ⇒ **`lwarpmk print`**

   The document is recompiled when a change is seen in the source. Several compilations may be necessary to resolve cross-references.

5. Force a recompile to occur.

   Enter ⇒ **`lwarpmk again`**

   Enter ⇒ **`lwarpmk print`**

   `lwarpmk again` updates the date code for the file, triggering a recompile the next time the document is made.[5]

6. Process the index.[6] [7]

   Enter ⇒ **`lwarpmk printindex`**

7. Recompile again to include the index.

   Enter ⇒ **`lwarpmk print`**

8. To force a single recompile when needed, even if no changes were detected:

   Enter ⇒ **`lwarpmk print1`**

Note that the HTML customization commands are ignored while making the print version.

[5] Although, when using the utility *latexmk* (introduced later), the changed date is ignored and an actual change in contents must occur to cause a recompile.

[6] The command **`lwarpmk printglossary`** is also available to process a glossary produced with the glossaries package. See section 8.6.12.

[7] Also see section 8.6.15 for index options.

5.3 Compiling the HTML version with *lwarpmk*

`lwarpmk html` is used to recompile an HTML version of the document.

⚠ Enable *lwarpmk*

1. If you have not yet done so, add `\usepackage{lwarp}` to the document, then compile the project a single time using *pdflatex*, *lualatex*, or *xelatex*. This generates the file `lwarpmk.conf`, which then allows the *lwarpmk* program to be used.

2. Compile the HTML version:

   Enter ⇒ **`lwarpmk html`**

   (a) *lwarpmk* uses LaTeX to process `tutorial_html.tex` to create `tutorial_html.pdf`.

   (b) *pdftotext* is then used to convert to the file `tutorial_html.html`. This file is a plain-text file containing HTML tags and content for the entire document.

   (c) *lwarpmk* manually splits `tutorial_html.html` into individual HTML files according to the HTML settings. For this tutorial, the result is `tutorial.html` (the home page), along with `First-chapter.html`[8], `Some-math.html`, and the document's index in `_Index.html`.[9]

3. View the HTML page in a web browser.

   Open the file `tutorial.html` in a web browser.

math images

   Note that math images have not yet been generated, so math is still displayed as its `alt` tag, which is set to the plain-text LaTeX source for that expression. Math may be displayed as SVG images (section 5.4) or by a MathJax script (section 5.5).

4. Force a recompile:

   Enter ⇒ **`lwarpmk again`**

   Enter ⇒ **`lwarpmk html`**

   Enter ⇒ **`lwarpmk print`**

5. Process the HTML index and recompile:[10][11]

   Enter ⇒ **`lwarpmk htmlindex`**

   Enter ⇒ **`lwarpmk html`**

   `_Index.html` is updated for the new LaTeX index.

6. Reload the web page to see the added index.

7. To force a single recompile when needed, even if no changes were detected:

   Enter ⇒ **`lwarpmk html1`**

[8] `First-chapter.html` also contains the first section, even though the second section is its own HTML page. This behavior is controlled by the boolean `CombineHigherDepths`.

[9] `index.html` is commonly used as a homepage, so the document index is in `_Index.html`.

[10] The command **`lwarpmk htmlglossary`** is also available to process a glossary produced with the glossaries package. See section 8.6.12.

[11] Also see section 8.6.15 for index options.

5.4 Generating the SVG images

math as SVG images

By default lwarp represents math as SVG images, with the LaTeX source included in `alt` attributes. In this way, the math is displayed as it was drawn by LaTeX, and the LaTeX source may be copied and pasted into other documents.

`picture` and Ti*k*Z

lwarp uses the same mechanism for `picture` and Ti*k*Z environments.

1. Create the SVG images:

   Enter ⇒ **`lwarpmk limages`**

   Enter ⇒ **`lwarpmk html`**

2. Move to the tutorial's HTML math page and reload the document in the browser.

3. The math images are displayed using the same font and formatting as the printed version.

4. Copy/paste a math expression into a text editor to see the LaTeX source.

⚠ adding/removing

When a math expression, `picture`, or Ti*k*Z environment is added or removed, the SVG images must be re-created by entering **`lwarpmk limages`** to maintain the proper image-file associations. Inline SVG math may be hashed and thus not need to be recreated, but display math and objects such as Ti*k*Z may move to new image numbers when the document is changed.

recompile first

Before attempting to create the SVG image files, *lwarpmk* verifies that the HTML version of the document exists and has correct internal image references.[12] If it is necessary to recompile the document's HTML version one more time, *lwarpmk* usually will inform the user with an error message, but there are some conditions which cannot be detected, so the user should watch for the LaTeX recompile warnings.

⚠ HTML instead of images

If HTML appears where an SVG image should be, recompile the document one more time to get the page numbers back in sync, then remake the images one more time.

⚠ page counter

Incorrect SVG images will also occur if the document changes the page counter:

```
\setcounter{page}{<value>}
```

The page counter must *not* be adjusted by the user.

⚠ Lots of files!

Expressing math as SVG images has the advantage of representing the math exactly as LaTeX would, but has the disadvantage of requiring an individual file for each math expression. For inline math, and some other objects, lwarp uses an MD5 hash on its LaTeX source to combine multiple instances of identical inline expressions into a single image file, but display math and other environments such as `picture` and Ti*k*Z require one image file each. For a document with a large amount of math, see section 5.5 to use MathJax instead.

[12] This becomes important when dealing with a document containing thousands of images.

5.5 Using MathJax for math

math with MathJax

Math may also be represented using the MathJax JavaScript project.

1. In the tutorial's source code, uncomment the `mathjax` package option for lwarp:

   ```
   mathjax, % Use MathJax to display math.
   ```

2. Recompile

   Enter ⇒ **`lwarpmk html`**

3. Reload the math page.

⚠ MathJax requirements

MathJax requires web access unless a local copy of MathJax is available, and it also requires that JavaScript is enabled for the web page. The math is rendered by MathJax. Right-click on math to see several options for rendering, and for copying the LaTeX source.

While using MathJax has many advantages, it may not be able to represent complex expressions or spacing adjustments as well as LaTeX, and it may not support some math-related packages.

5.6 Changing the css style

For a formal css style, add to the preamble:

```
\usepackage{lwarp}
. . .
\CSSFilename{lwarp_formal.css}
. . .
\begin{document}
```

For a modern css style, `lwarp_sagebrush.css` is also provided:

```
\CSSFilename{lwarp_sagebrush.css}
```

See section 7.7 for more information about modifying the css styling of the document.

5.7 Customizing the HTML output

A number of settings may be made to control the HTML output, including filename generation, automatic compilation, math output, document splitting, meta data, and page headers and footers.

See section 7.6 for more information.

5.8 Using *latexmk*

latexmk is a LaTeX utility used to monitor changes in source files and recompile as needed.

1. In the tutorial's source code uncomment the `latexmk` option for the lwarp package:

   ```
   latexmk, % Use latexmk to compile.
   ```

2. Recompile the printed version of the document.

   Enter ⇒ **`lwarpmk print`**

   lwarp updates its own configuration files (`lwarpmk.conf` and `tutorial.lwarpmkconf`) whenever the printed version of the document is compiled. These configuration files remember that *lwarpmk* should use *latexmk* to compile the document.

3. Recompile the document.

   Enter ⇒ **`lwarpmk print`**

   and/or

   Enter ⇒ **`lwarpmk html`**

Changes are detected by comparing checksums rather than modification times, so `lwarpmk again` will not trigger a recompile, but *latexmk* has a much better awareness of changes than the *lwarpmk* utility does and it is likely to correctly know when to recompile. A recompile may be forced by making a small change to the source, and a single recompile may be forced with:

forced single-pass recompile

Enter ⇒ **`lwarpmk print1`**

and/or

Enter ⇒ **`lwarpmk html1`**

5.9 Using XeLaTeX or LuaLaTeX

XeLaTeX or LuaLaTeX may be used instead of LaTeX.

1. Remove the auxiliary files for the project:

   Enter ⇒ **`lwarpmk cleanall`**

2. Use *xelatex* or *lualatex* to compile the printed version a single time.

   Enter ⇒ **`xelatex tutorial.tex`**

   — *or* —

   Enter ⇒ **`lualatex tutorial.tex`**

   When the compile occurs, the configuration files for *lwarpmk* are modified to remember which TeX engine was used. XeLaTeX or LuaLaTeX will be used for future runs of *lwarpmk*.

3. To recompile the document:

   Enter ⇒ **`lwarpmk print`**

   -and-

   Enter ⇒ **`lwarpmk html`**

4. Also remember to update the indexes and recompile again:

   Enter ⇒ **`lwarpmk htmlindex`**

   Enter ⇒ **`lwarpmk html`**

   Enter ⇒ **`lwarpmk printindex`**

   Enter ⇒ **`lwarpmk print`**

5.10 Using DVI LaTeX

Traditional DVI LaTeX may also be used along with `.eps` image files. An SVG version of each image must also be provided. *lwarpmk* may be used to convert image formats.

To convert EPS files to PDF:

Enter ⇒ **`lwarpmk epstopdf *.eps`** **(or a list of files)**

To convert PDF files to SVG:

Enter ⇒ **`lwarpmk pdftosvg *.pdf`** **(or a list of files)**

⚠ bitmapped fonts See section 7.4 regarding font selection to avoid the use of bitmapped fonts.

5.11 Using a glossary

lwarp supports the gloss and glossaries packages, although this tutorial does not supply an example.

5.11.1 gloss package

See section 8.6.11.

5.11.2 glossaries package

To process the glossary for the print version:

Enter ⇒ **`lwarpmk printglossary`**

⚠ (If **`makeglossaries`** is not found, see section 8.6.12.)

To process the glossary for the HTML version:

Enter ⇒ **`lwarpmk htmlglossary`**

In each case, the document will have to be recompiled afterwards:

Enter ⇒ **`lwarpmk html1`**

Enter ⇒ **`lwarpmk html`**

Enter ⇒ **`lwarpmk print1`**

Enter ⇒ **`lwarpmk print`**

See section 8.6.12 to set options for processing glossaries.

5.12 Cleaning auxiliary files

To remove the auxiliary files `.aux`, `.toc`, `.lof`, `.lot`, `.idx`, `.ind`, `.log`, and `.gl*`, and a few others:

Enter ⇒ **`lwarpmk clean`**

5.13 Cleaning auxiliary and output files

To remove the auxiliary files, and also remove the `.pdf` and `.html` files:

Enter ⇒ **`lwarpmk cleanall`**

5.14 Cleaning the images from the <project>-images directory

The `<project>-images` directory contains SVG images automatically generated for inline and display math, tikz, etc. To remove all the images from the `<project>-images` directory:

Enter ⇒ **`lwarpmk cleanlimages`**

5.15 Converting PDF or EPS images to SVG

HTML cannot display PDF or EPS images, so any external PDF graphics images must be converted to SVG format. *pdftocairo* and *epstopdf* may be used one image at a time, but *lwarpmk* also provides a way to convert PDF or EPS images in bulk:

Enter ⇒ **`lwarpmk epstopdf *.eps`** **(or a list of files)**

Enter ⇒ **`lwarpmk pdftosvg *.pdf`** **(or a list of files)**

Be sure to always provide SVG files for HTML output.

5.16 Creating HTML from an incomplete compile

During testing it may be useful to finish the HTML conversion even when the document had errors and did not compile successfully. To attempt an HTML conversion of an incomplete document:

Enter ⇒ **`lwarpmk pdftohtml [-p project]`**

5.17 Processing multiple projects in the same directory

⚠ xr, xr-hyper, xcite

It is possible to have several projects in the same directory. *lwarpmk* has an optional parameter which is the document to compile.

To create each project:

Enter ⇒ **`pdflatex project_a`**

Enter ⇒ **pdflatex project_b**

Each project is given its own configuration file:

`project_a.lwarpmkconf, project_b.lwarpmkconf`

To compile each project with `lwarkmk`:

Enter ⇒ **lwarpmk print -p project_a**

Enter ⇒ **lwarpmk print -p project_b**

Enter ⇒ **lwarpmk html -p project_a**

Enter ⇒ **lwarpmk html -p project_b**

To generate each project's images:

Enter ⇒ **lwarpmk limages -p project_a**

Enter ⇒ **lwarpmk limages -p project_b**

To clean each project's images:

Enter ⇒ **lwarpmk cleanlimages -p project_a**

Enter ⇒ **lwarpmk cleanlimages -p project_b**

To clean each project's auxiliary files:

Enter ⇒ **lwarpmk cleanall -p project_a**

Enter ⇒ **lwarpmk cleanall -p project_b**

If using *bibtex*, for example, the HTML version must also be processed:

Enter ⇒ **bibtex project_a_html**

5.18 Using the *make* utility

lwarpmk has an action which may be useful for integration with the common *make* utility:

`lwarpmk pdftohtml [-p project]`

make may be used to compile the code to PDF with HTML tags (`project_html.pdf`), then *lwarpmk* may be used to convert each target to HTML files.

5.19 What next?

How do I do something? See the General Index.

Something do not work! See the Troubleshooting Index or section 13: Troubleshooting.

Package options: See section 29, Package options.

HTML and filename settings: See section 7.6, Customizing the HTML output.

Footnote placement: See section 7.6, Customizing the HTML output.

Title page, indexing, glossaries: See section 8.6, Front and back matter.

Shell escape: See section 7.3, Shell escape.

CSS customization: See section 7.7, Customizing the CSS.

MathJax customization: See section 8.7.7, Customizing MathJax.

Localization: (languages) — See section 7.1, Localization.

Accessibility: (`alt` and `title` tags) — See section 7.2, Accessibility.

Converting an existing document: See section 6, Converting an existing document.

EPUB conversion: See section 10, EPUB conversion.

Word processor conversion: See section 11, Word-processor conversion.

6 Converting an existing document

To convert an existing document for use with lwarp:

1. Arrange the document in the following order:
   (a) Declare the `\documentclass`.
   (b) Load text fonts.
   (c) Load inputenc or inputenx, fontenc, or fontspec.
   (d) Load lwarp.
   (e) Load remaining packages.
2. Modify the document:

⚠ math in section names

   (a) If using named HTML files, in section names use paren math `\(x+y\)` instead of dollar math `$x+y$`. (Dollar math works, but appears in the filename.) Or, use a short name for the TOC entry without the math, or use `\texorpdfstring` from the hyperref package:

```
\section{Some math \texorpdfstring{$1+2=3$}{three}}
```

⚠ scale

   (b) Avoid using the `\includegraphics` `scale` option. Change:

```
\includegraphics[scale=<xx>]{ . . . }
```

   to:

```
\includegraphics[width=<yy>\linewidth]{ . . . }
```

⚠ tabular

   (c) Possible changes to `tabular` environments include: `*` columns, multirow, longtable, supertabular, xtab, bigdelim. See section 8.10.1.

⚠ package options

   (d) If using braces in package options, such as with caption, see section 8.1.
   (e) Possible option clashes with memoir. See section 8.13.

⚠ indexes

   (f) If using indexes, see section 8.6.15.
   (g) If using many indexes, glossaries, `.aux` files, etc., see section 8.6.15 regarding morewrites. If morewrites is already used, be sure to add the setup with `allocate=10`.
   (h) Other changes as per Special cases and limitations, section 8.
3. Convert any PDF images to SVG. See section 8.8.
4. Manually compile the print version with *latex*, *pdflatex*, *lualatex*, or *xelatex*.
5. `lwarpmk print` to finish the print version.
6. `lwarpmk html` to create the HTML version.
7. `lwarpmk limages` to create the SVG images of any SVG math, `lateximage`, Ti*k*Z, etc.

Need help?

See the General Index for "how-to", and the Troubleshooting Index if something doesn't work. A Troubleshooting section is also available. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category.

Table 5: Localization settings

Object names: LaTeX provides redefinable names for various objects, and lwarp adds a few more. Use `\renewcommand` to change these.

`\abstractname`: This macro is honored by lwarp.

`\linkhomename`: Displayed by the link to the homepage.

`\linkpreviousname`: Displayed by the link to the previous page

`\linknextname`: Displayed by the link to the next page.

`\sidetocname`: Displayed at the head of the sidetoc.

HTML settings: See table 8 and section 7.6 for details.

`\HTMLLanguage`: The language to declare for each web page.

`\ImageAltText`, `\MathImageAltText`, `\PackageDiagramAltText`, `\AltTextOpen`, `\AltTextClose`: The defaults used for HTML `alt` text for images. See section 7.2.

`\CSSFilename`: The name of the css file to use.

`\MathJaxFilename`: The name of the MathJax script to use.

Package options:

`ImagesName` and `ImagesDirectory`: These options control the filenames used by lwarp when it automatically generates images. See table 7 and section 7.5.

`xindyStyle`, `xindyLanguage`, `xindyCodepage`: When using *xindy*, these options may be set according to local use. See section 8.6.21.

`pdftotextEnc`: To adjust the encoding of *pdftotext*.

7 Additional details

7.1 Localization

Regional localization is supported by lwarp via the package options and macros shown in table 5.

7.2 Accessibility

lwarp provides several methods for improving access to the document using tools such as text-only browsers, copy / paste, text-to-speech readers, or Braille readers. lwarp can use the HTML `alt` text attribute for images, as describe below. lwarp can also use the HTML `title` attribute, which usually generates a pop-up text. lwarp can add this to a reference or hyperlink. lwarp also uses standard HTML5 elements which are pre-assigned ARIA roles for increased accessibility, and lwarp assigns the `math` role for SVG math images, and the `note` role for footnotes, end notes, margin paragraphs and notes, etc. MathJax also has provisions for improved accessibility as well. See table 6.

Table 6: Accessibiltiy settings

\ImageAltText: The default HTML `alt` text for `\includegraphics` and `lateximages`. Set with `\renewcommand`.

\includegraphics alt key: For `\includegraphics`, lwarp adds the `alt` key/value. For example:

```
\includegraphics[alt={Some text.}]{filename}
```

SVG math: For simple SVG math, lwarp places the LaTeX math expression in the `alt` text, so that the LaTeX expression may be copied and pasted to another document as plain text.

\MathImageAltText: For complicated SVG math, such as enclosed in `\InlineMathOther`/`\InlineMathNormal`, or `\DisplayMathOther`/`\DisplayMathNormal`, the HTML `alt` text will be set to `\MathImageAltText`. Set with `\renewcommand`.

MathJax: For MathJax, the accessibility tools provided by MathJax are enabled by default by lwarp's MathJax scripts.

\PackageDiagramAltText: Various packages create diagrams which lwarp converts into SVG images. These are given `alt` text set to `\PackageDiagramAltText`. Set with `\renewcommand`.

\ThisAltText: The HTML `alt` text of the next image may be set with:

```
\ThisAltText{Custom text about the image.}
<SVG math, Tikz, picture, etc.>
```

The next single image will be generated with the given text, and the following images will revert to back to their defaults.

`\ThisAltText` may also be used to assign an HTML `title` to the next reference or hyperlink.

```
\ThisAltText{Custom text about the link.}
Text ... \ref{label_name} ... text.
```

See section 7.6.

\AltTextOpen and \AltTextClose: By default, HTML `alt` text is enclosed by parentheses. This may be changed by redefining `\AltTextOpen` and `\AltTextClose`. Set with `\renewcommand`.

7.3 Shell escape

`-\/-shell-escape` (*Opt*)

Some documents require the use of an external program, which is allowed when using the `--shell-escape` command-line option. When the document is first compiled manually, and also whenever the print version is recompiled, lwarp detects and remembers whether shell escape is enabled. If so, it will also be enabled when the document is recompiled with *lwarpmk*.

7.4 Font and UTF-8 support

⚠ type 3 bitmapped fonts

lwarp uses *pdftotext* to convert PDF output into UTF-8-encoded text. This process requires that UTF-8 information be embedded in the PDF file, which may prevent the use of older "type 3" bit-mapped fonts, and of older packages such as ae. The lwarp option `pdftotextEnc` may be useful in some situations. See section 7.5.

vector fonts
Computer Modern
⚠ *pdflatex*
⚠ DVI *latex*
`cm-super` (*Pkg*)

While using DVI *latex* or PDF *pdflatex*, if no font-related package is specified then the default Computer Modern font is used, which may be a "type 3" bit-mapped font which may not convert well to plain text. A "type 1" vector font is required.

To use the updated cm-super's type 1 fonts instead of Computer Modern, install the cm-super font package.

`lmodern` (*Pkg*)

To use Latin Modern instead, add

```
usepackage{lmodern}
```

to the preamble.

`dejavu` (*Pkg*)

Another useful option is the Deja Vu series of fonts, which have an increased coverage of language and glyphs:

```
\usepackage{dejavu}
```

latex, *pdflatex*, T1, UTF8

While using DVI *latex* or PDF *pdflatex*, lwarp automatically loads fontenc with T1 encoding. fontenc may be loaded with an additional encoding after lwarp. inputenc is automatically loaded with UTF8 encoding if if has not yet been loaded, but may also be specified with another encoding such as `latin1`. See the next section regarding index encoding.

⚠ *xelatex*, *lualatex*, fontspec

XeLaTeX and LuaLaTeX users must use the fontspec package. Do NOT use fontenc!

Place fontspec or fontenc, xunicode, and other font and UTF-8 related commands after the `\documentclass` command and before `\usepackage{lwarp}`.

⚠ package conflicts

In some cases, a package conflict may require that a font package be loaded after lwarp, which should work as well:

1. `documentclass{article/book/report}` comes first, followed by any of:
2. Font and UTF-8 related commands:
   - For XeLaTeX or LuaLaTeX:
     - fontspec and font choices

`fontspec` (*Pkg*)

ligatures

lwarp sets the following to turn off TeX ligatures during the generation of HTML tags, and turn off common ligatures in regular text,

since older browsers may not display them correctly and newer browsers can automatically re-create them.

```
\defaultfontfeatures[\rmfamily]{Ligatures={NoCommon,TeX}}
\defaultfontfeatures[\sffamily]{Ligatures={NoCommon,TeX}}
\defaultfontfeatures[\ttfamily]{Ligatures=NoCommon}
```

- For *pdflatex*:
  - (a) `\usepackage{lmodern}`, or other font-related packages (`lmodern` (*Pkg*))
  - (b) `\usepackage[T1]{fontenc}` (`fontenc` (*Pkg*))
  - (c) `\usepackage[utf8]{inputenc}`, or `latin1`, etc. Or use inputenx. (`inputenc` (*Pkg*), `inputenx` (*Pkg*))
  - (d) `\usepackage{newunicodechar}` along with related definitions. (`newunicodechar` (*Pkg*))
  - (e) To assist with the PDF-HTML conversion: (`glyphtounicode.tex` (*file*))
    i. `\input glyphtounicode.tex`
    ii. `\input glyphtounicode-cmr.tex% from the pdfx package`
    iii. `\pdfgentounicode=1`
  - (f) Another option to assist with the PDF-HTML conversion, such as the dotless j (`\j`): (⚠ dotless j)
    - `\usepackage{cmap}` — *or* —
    - `\usepackage{mmap}` — *or* —
    - `\usepackage[noTeX]{mmap}`

    (`cmap` (*Pkg*), `mmap` (*Pkg*))
  - (g) `\usepackage{textcomp}` (`textcomp` (*Pkg*))

3. `\usepackage{newtxmath}` or other math-related font packages. Many of these load amsmath, which may now be loaded before lwarp.
4. `\usepackage{lwarp}` (section 7.5) is placed after any of the above, followed by:
5. `\setmonofont{TeX Gyre Cursor}` or similar may be required if using XeLaTeX or LuaLaTeX and fontspec along with traditional font packages such as txfonts, newtxtext, etc. This is required to turn off the monospaced font's ligatures with fontspec after loading the traditional font packages. Monospaced output ligatures must be turned off to produce the correct HTML characters. (⚠ fontspec with monospaced fonts)

   Any monospace font with built-in ligatures may require these ligatures to be disabled for HTML. In one example, JetBrain Mono, it is required to use (⚠ JetBrain Mono ⚠ HTML corrupted)

   ```
   \setmonofont{JetBrains Mono}[%
   . . .
   Contextuals=AlternateOff,
   ]
   ```

   After lwarp is loaded, the ligature may be re-enabled for print mode by using `\setmonofont` again inside a `warpprint` environment.
6. … the rest of the preamble and the main document.

⚠ UTF-8 locale In some cases, an external program may require a UTF-8 "locale". See section 9.9.

7.4.1 Indexes, glossaries, and encoding

lwarp supports *makeindex*, *xindy*, *xindex*, and glossaries, gloss, and nomencl.

See section 8.6.14 for indexing, and section 8.6.12 for the glossaries package.

7.5 lwarp package loading and options

lwarp supports book, report, and article classes, as well as the equivalent Koma-script classes and memoir, and various CJK-related classes and packages.

Load the lwarp package immediately after the font and UTF-8 setup commands.

Package options may be set while loading lwarp, or later with

```
\lwarpsetup{⟨key=value, . . . ⟩}
```

lwarp (*Pkg*) lwarp package options are as follows:

mathsvg (*Opt*)
mathjax (*Opt*)
Default: mathsvg

mathsvg and **mathjax:** Selects SVG images or MathJax for math display. See section 8.7.

latexmk (*Opt*)
Default: false

latexmk: Tells *lwarpmk* to use *latexmk* to recompile the document several times if necessary. Otherwise, *lwarpmk* attempts to determing for itself whether to recompile. See section 7.6.

dvips (*Opt*)
Default: false

dvips: Tells *lwarpmk* to use *dvips* and *ps2pdf* to convert DVI output to PDF.

dvipdfm (*Opt*)
Default: false

dvipdfm: Tells *lwarpmk* to use *dvipdfm* to convert DVI output to PDF.

dvipdfmx (*Opt*)
Default: false

dvipdfmx: Tells *lwarpmk* to use *dvipdfmx* to convert DVI output to PDF.

HomeHTMLFilename (*Opt*)
Default: \BaseJobname

HomeHTMLFilename:

Filename of the homepage, without the ".html" suffix. Defaults to the \BaseJobname. A common setting is:

```
HomeHTMLFilename=index
```

filename underscores

causing the homepage to be the file index.html. Underscores are allowed in HomeHTMLFilename and HTMLFilename options, but may need to be escaped elsewhere, such as when appearing in a list:

```
\item [\href{file\_name.pdf}{text}] \
```

See section 7.6.1 for examples of naming and numbering HTML files.

HTMLFilename (*Opt*)
Default: <empty>

HTMLFilename: A filename prefix for the rest of the HTML web pages. Useful for numbered web pages with a common prefix. May be empty. See section 7.6.1 for examples of naming and numbering HTML files.

ImagesName (*Opt*)
Default: image-

ImagesName: The prefix for the images automatically generated by lwarp for objects such as SVG math and lateximages.

ImagesDirectory (*Opt*)
Default: \jobname-images

ImagesDirectory: The directory for the images automatically generated by lwarp for objects such as SVG math and lateximages. By default, these images will appear in a directory named <jobname>-images, and the images will be named and numbered image-<nn>.

Table 7: Lwarp package options

Option	Description
mathsvg	Show math using SVG images.
mathjax	Show math using MathJax.
latexmk	Use *latexmk* for compiling documents.
dvips	Use *dvips* and *ps2pdf* to convert DVI documents.
dvipdfm	Use *dvipdfm* to convert DVI documents.
dvipdfmx	Use *dvipdfmx* to convert DVI documents.
HomeHTMLFilename	The filename of the home page.
HTMLFilename	A prefix for the filenames of the remaining web pages.
ImagesName	A prefix for the filenames of generated images.
ImagesDirectory	The directory used to hold generated images.
PrintLatexCmd	The shell commands for **lwarpmk print**.
HTMLLatexCmd	The shell commands for **lwarpmk html**.
For indexing (section 8.6.15) and glossaries (section 8.6.12):	
makeindex	Use *makeindex* to generate indices.
makeindexStyle	Set a custom style for *makeindex*.
xindy	Use *xindy* to generate indices.
xindyStyle	Set a custom style for *xindy*.
xindyLanguage	The *xindy* language option used for index generation.
xindyCodepage	The *xindy* codepage option used for index generation.
xindex	Use *xindex* to generate indices.
xindexConfig	Set a custom configuration file for *xindex*.
PrintIndexCmd	Shell commands executed by **lwarpmk printindex**.
HTMLIndexCmd	Shell commands executed by **lwarpmk htmlindex**.
LatexmkIndexCmd	Shell commands executed by *latexmk*.
IndexRef	How to format index links.
GlossaryCmd	Shell command executed by **lwarpmk printglossary** and **lwarpmk htmlglossary**.
Seldom necessary:	
OSWindows	Force compatibility with MS-Windows.
pdftotextEnc	Set the encoding for *pdftotext*.
lwarpmk	Generate a local copy of `lwarpmk.lua`.
Used internally by lwarp:	
warpprint	Generate print output, and also generate configuration files.
warpHTML	Generate HTML output.
BaseJobname	The `\jobname` to use. Set to the `\jobname` of the printed version even while generating HTML.
warpdisable	Disables most of lwarp for testing purposes.

PrintLatexCmd (*Opt*)
Default: <automatic>

PrintLatexCmd: Sets the shell commands executed by **lwarpmk print**. If not specified, will automatically be set according to the detected LaTeX engine and the use of --shell-escape.

HTMLLatexCmd (*Opt*)
Default: <automatic>

HTMLLatexCmd: Sets the shell commands executed by **lwarpmk html**. If not specified, will automatically be set according to the detected LaTeX engine and the use of --shell-escape.

makeindex (*Opt*)
Default: makeindex

makeindex: Sets PrintIndexCmd, HTMLIndexCmd, and LatexmkImageCmd to use *makeindex* when generating indexes with **lwarpmk printindex**, **lwarpmk htmlindex**, or *latexmk*. If neither makeindex nor xindy is used, makeindex is assumed.

makeindexStyle (*Opt*)
Default: lwarp.ist

makeindexStyle: If you wish to use a custom .ist file for index generation, see section 8.6.20.

xindy (*Opt*)
Default: makeindex

xindy: Sets PrintIndexCmd, HTMLIndexCmd, and LatexmkImageCmd to use *xindy* when generating indexes with **lwarpmk printindex**, **lwarpmk htmlindex**, or *latexmk*.

xindyStyle (*Opt*)
Default: lwarp.xdy

xindyStyle: If you wish to use a custom .xdy file for index generation, see section 8.6.21.

xindyLanguage (*Opt*)
Default: english

xindyLanguage: If using an index or glossary, see section 29.

xindyCodepage (*Opt*)
Default: utf8

xindyCodepage: If using an index, see section 29.

xindex (*Opt*)
Default: makeindex

xindex: Sets PrintIndexCmd, HTMLIndexCmd, and LatexmkImageCmd to use *xindex* when generating indexes with **lwarpmk printindex**, **lwarpmk htmlindex**, or *latexmk*.

xindexConfig (*Opt*)
Default: <empty>

xindexConfig: If you wish to use a custom xindex-*.lua file for index generation, see section 8.6.22.

PrintIndexCmd (*Opt*)
Default: <automatic>

PrintIndexCmd: Sets the shell commands executed by **lwarpmk printindex**. If not specified, will be set by the selection of makeindex or xindy. May be used to specify the creation of multiple indexes. See section 8.6.15.

Examples:

```
makeindex -s lwarp.ist projectname.idx              (makeindex)
xindy -M lwarp.xdy -L english -C utf8 projectname.idx   (xindy)
```

automatic setting

The use of the makeindex or xindy options sets PrintIndexCmd to sensible values for each of those programs while compiling a single index. lwarp's makeindexStyle, xindyStyle, xindyLanuage, and xindyCodepage options will be used if specified.

⚠ xindy

If specifying PrintIndexCmd manually, be sure to assign an *xindy* language and codepage with the -L and -C *xindy* options, as the lwarp xindyLanguage and xindyCodepage options are not used for the PrintIndexCmd option when it is set manually.

This option is stored in the configuration files `lwarpmk.conf` and `*.lwarpmkconf`, and is then passed by the **`lwarpmk printindex`** command to the operating system to compile the print indexes. Since the command string is parsed by TEX, written to a file, read from the file by LuaTEX, and finally passed to the operating system, any attempt at quoting will be problematic. For complicated commands, it would be best to create a shell script, and simply refer to the script with the lwarp `PrintIndexCmd` option.

`HTMLIndexCmd` (*Opt*)
Default: `<automatic>`

HTMLIndexCmd: Sets the shell commands executed by **`lwarpmk htmlindex`**. If not specified, will be set by the selection of `makeindex` or `xindy`. May be used to specify the creation of multiple indexes. See section 8.6.15.

⚠ filenames

Example settings are similar to `PrintIndexCmd`, but append `_html` to the filenames:

```
makeindex -s lwarp.ist projectname_html.idx        (makeindex)
xindy -M lwarp.xdy -L english -C utf8 projectname_html.idx
(xindy)
```

automatic setting

The use of the `makeindex` or `xindy` options sets `HTMLIndexCmd` to sensible values for each of those programs while compiling a single index. lwarp's `makeindexStyle`, `xindyStyle`, `xindyLanuage`, and `xindyCodepage` options will be used if specified.

⚠ xindy

If specifying `HTMLIndexCmd` manually, be sure to assign an *xindy* language and codepage with the `-L` and `-C` *xindy* options, as the lwarp `xindyLanguage` and `xindyCodepage` options are not used for the `HTMLIndexCmd` option when it is set manually.

As with `PrintIndexCmd`, to generate complicated indexes it may be worthwhile to use a shell script, then refer to that script with `HTMLIndexCmd`.

`LatexmkIndexCmd` (*Opt*)
Default: `<automatic>`

LatexmkIndexCmd: Sets the shell commands executed by *latexmk*. Unlike `PrintIndexCmd` and `HTMLIndexCmd`, `LatexmkIndexCmd` does not include any filenames, which will be provided instead by *latexmk*. See section 8.6.15.

Example settings are similar to `PrintIndexCmd`, but without a filename:

```
makeindex -s lwarp.ist                          (makeindex)
xindy -M lwarp.xdy -L english -C utf8              (xindy)
```

automatic setting

The use of the `makeindex` or `xindy` options sets `LatexmkIndexCmd` to either of the two settings show above. lwarp's `makeindexStyle`, `xindyStyle`, `xindyLanuage`, and `xindyCodepage` options will be used if specified. Unlike `PrintIndexCmd` and `HTMLIndexCmd`, *latexmk* uses either of the single-line settings of `LatexmkIndexCmd` shown above to compile each of multiple indexes if necessary.

⚠ xindy

If specifying `LatexmkIndexCmd` manually, be sure to assign an *xindy* language and codepage with the `-L` and `-C` *xindy* options, as the lwarp `xindyLanguage` and `xindyCodepage` options are not used for the `LatexmkIndexCmd` option when it is set manually.

`IndexRef` (*Opt*)
Default: `cref`

IndexRef: Describes how to display the index entries for HTML output. Possible values are `ref`, `nameref`, `refnameref`, `cref`, `crefnameref`, `autoref`, or a text string such as `(link)` or `(*)` for each index entry reference. (Adding parentheses around a single character makes the link larger and easier to click on.) The default is `cref`, which is available even if the print document does

not use cleveref, as the lwarp package relies on cleveref during HTML output. Option `autoref` gives the same results as `cref`.

`\ref` and `\cref` to starred or otherwise unknown links will display as `(*)` instead of ??.

⚠ ?? If using `cref` (the default), and if a reference appears as ?? with a non-functional link, use cleveref's `\crefname` to give a name to that type of label.

In general, `crefnameref` gives the most information, but the index can become quite verbose. Using `(*)` or similar yields a very compact index.

`GlossaryCmd` (*Opt*)
Default: **`makeglossaries`**

GlossaryCmd: Sets the shell command executed by **`lwarpmk printglossary`** and **`lwarpmk htmlglossary`**. The print or HTML glossary filename is appended to this command. See section 8.6.12.

`OSWindows` (*Opt*)

OSWindows: lwarp attempts to automatically sense WINDOWS, but it may be forced with this option. See section 7.9.

`pdftotextEnc` (*Opt*)
Default: `UTF-8`

pdftotextEnc: Used to specify the encoding used by *pdftotext* during the PDF-HTML conversion. In most situations, the default is the correct choice.

`lwarpmk` (*Opt*)

lwarpmk: If you wish to have lwarp generate a local copy of `lwarpmk.lua` for archival or local-installation purposes, compile the print version with the `lwarpmk` option set. See section 29.

The following options are used internally by lwarp, and usually are not used in the user's document:

`warpprint` (*Opt*)
`warpHTML` (*Opt*)

warpprint and **warpHTML:** Usually controlled by *lwarpmk*, and not set in the document. Select the `warpprint` option to generate print output (default), or the `warpHTML` option to generate HTML5 output. The default is print output, so the print version may be compiled with the usual *pdflatex*, etc. When lwarp is loaded in print mode, it creates `<project>_html.tex`, which sets the `warpHTML` option before calling the user's source code `<project>.tex`. In this way, `<project>.tex` can `\usepackage{lwarp}` without any options to create a printed version, while `<project>_html.tex` will create an HTML version.

`BaseJobname` (*Opt*)
Default: `\jobname`

BaseJobname: Not intended for the user. Used internally by lwarp when creating the `*_html.tex` file used to compile the HTML version. See section 29.

`warpdisable` (*Opt*)

warpdisable: Internally disables both `warpprint` and `warpHTML`. This disables most of lwarp, which may be useful for testing purposes to see whether lwarp is causing a problem.

7.6 Customizing the HTML output

Table 8 shows several settings may be used to customize the HTML output. Watch for the correct placement of each!

⚠ Placement!

⚠ Changes! Note that if changes are made, it is best to first:

1. Clear all the HTML, PDF, and auxiliary files:

   Enter ⇒ **`lwarpmk cleanall`**

2. Recompile the print version in order to recreate the configuration files for *lwarpmk*:

   Enter ⇒ **`lwarpmk print`**

3. Finally, recompile the HTML version with the new settings:

   Enter ⇒ **`lwarpmk html`**

Placed in the preamble before `\begin{document}`:

`\HTMLFirstPageTop` Default: `<empty>`

`\HTMLFirstPageTop`: {⟨*contents*⟩} A user-definable custom action applied to the top of the home page. Useful for logos, etc. `\LinkNext` may be used to link to the next web page. Defaults empty. Ignored in print output.

`\HTMLFirstPageBottom` Default: `<empty>`

`\HTMLFirstPageBottom`: {⟨*contents*⟩} A user-definable custom action applied to the bottom of the home page. Useful for logos, etc. `\LinkNext` may be used to link to the next web page. Defaults empty. Ignored in print output.

`\linkhomename` Default: `Home`

`\linkhomename`: Name of the link to the home page. Paragraphs are allowed. Redefine with `\renewcommand`.

`\linkpreviousname` Default: `Previous`

`\linkpreviousname`: Name of the link to the previous page. Paragraphs are allowed. Redefine with `\renewcommand`.

`\linknextname` Default: `Next`

`\linknextname`: Name of the link to the next page. Paragraphs are allowed. Redefine with `\renewcommand`.

`tocdepth` (*Ctr*)

`tocdepth`: Sectioning depth of the table of contents. See section 16 for a list of LaTeX stack depths.

`SideTOCDepth` (*Ctr*) Default: `1`

`SideTOCDepth`: Sectioning depth of the sideTOC. Defaults to 1, causing the sideTOC to show sections but not subsections.

sideTOC

Each subpage of the website has its own small table of contents on the side (the "sideTOC"). Its depth is set by `SideTOCDepth`. This sideTOC is only shown if the browser display is wide enough. When using a narrow web browser window, "responsive web design" is used to show the sideTOC at the top of the page, as well as a link back to **Home** at the top and bottom.

It is recommended to set:

```
SideTOCDepth = FileDepth
```

or

Table 8: HTML settings

Macro/Cntr/Bool	Loc*	Description
\linkhomename	P	Name of the link to the homepage.
\linkpreviousname	P	Name of the link to the previous page.
\linknextname	P	Name of the link to the next page.
SideTOCDepth	P	Sectioning depth of the sideTOC.
\sidetocname	P	Name of the sideTOC.
FileDepth	P	Sectioning depth of the file splits.
CombineHigherDepths	P	Combine higher section levels.
FileSectionNames	P	Use section names for file names, else use numbers.
\FilenameLimit	P	Maximum length of the generated filenames.
FootnoteDepth	P	Sectioning depth of footnotes.
\abstractname	P	The name of the abstract.
\ImageAltText	PD	\includegraphics and other images' alt tag.
\ThisAltText {⟨text⟩}	PD	Assigns an alt/title tag for the next image or link.
\MathImageAltText	PD	The SVG math image lateximage alt tag.
\PackageDiagramAltText	PD	The suffix for a package's lateximage alt tags.
\AltTextOpen	PD	Start an HTML alt tag.
\AltTextClose	PD	End an HTML alt tag.
\CSSFilename	PS	The CSS for the following files.
\MathJaxFilename	PS	The MATHJAX script for the following files.
\HTMLLanguage	PS	The HTML lang tag.
\HTMLTitle	PS	The homepage's <title>, overriding \title.
\HTMLTitleBeforeSection	PS	Set subpage <title>s to \HTMLTitle - sectionname
\HTMLTitleAfterSection	PS	Set subpage <title>s to sectionname - \HTMLTitle
\HTMLAuthor	PS	The HTML author meta tag, overriding \author.
\HTMLDescription	PS	The HTML description meta tag.
\HTMLFirstPageTop	P	Heading for the home page.
\HTMLFirstPageBottom	P	Footer for the home page.
\HTMLPageTop	PS	Heading for the other pages.
\HTMLPageBottom	PS	Footer for the other pages.
\HTMLnewcolumntype	D	\newcolumntype for HTML.
\IndexPageSeparator	P	Index page list separator.
\IndexRangeSeparator	P	Index page range separator.
FixSmallCaps	P	Set true if small caps rendered as all caps.
HTMLDebugComments	P	Boolean to generate HTML comments.

* **P:** Preamble, **D:** Anywhere in the document. **S:** Before a section.

SideTOCDepth = FileDepth+1

⚠ inaccessible pages

If SideTOCDepth < FileDepth, web pages will be inaccessible via the sideTOC.

\sidetocname
Default: Contents

\sidetocname: Name of the sideTOC. Paragraphs are allowed. Redefine with \renewcommand.

FileDepth (*Ctr*)
Default: -5

FileDepth: Sectioning depth of file splits. Defaults to -5, causing the entire HTML website to be one single file.

- To place the entire file into one HTML page, use:
  \setcounter{FileDepth}{-5}
- To split the HTML file at \section depth, use:
  \setcounter{FileDepth}{1}
- ⚠ To ensure that the HTML pages/files are accessible:
  Place a \tableofcontents somewhere before the first section break (therefore in the "home page"), and set
  tocdepth >= FileDepth

CombineHigherDepths (*bool*)
Default: true

CombineHigherDepths: Combine a higher section with its first lower subsections, down to the FileDepth. Defaults to true. Set to false to simulate the concept of a chapter opening on its own page, for example.

The file splits are controlled by the counter FileDepth and the boolean CombineHigherDepths. Setting FileDepth to 0 splits the file at chapters, 1 at sections, etc. CombineHigherDepths controls whether to combine pages at levels higher than the chosen FileDepth, such as in this tutorial where the page which opens the chapter also contains the first section. Be careful to set tocdepth and SideTOCDepth to allow access to each page of the website. Set tocdepth and SideTOCDepth to be greater than or equal to FileDepth.

⚠ Inaccesible pages!

⚠ Lost in an old page!

When making changes to the file structure, it is possible to end up with the web browser pointing to an old file which is no longer in use. When this occurs, changes to the web site will not appear in the browser, even if reloading the page, because that page is no longer in use. It is best to return to the home page, clean the files (lwarpmk cleanall), change FileDepth and/or CombineHigherDepths, then finally recompile and renavigate to the desired page using the new file structure.

FileSectionNames (*bool*)
Default: true

FileSectionNames: If true, web page filenames are derived from a sanitized version of the section names. If false, web pages are numbered. Either way, the HTMLFilename option is used as a prefix. See section 7.6.1 for examples of naming and numbering HTML files. The user must ensure that filenames are unique after begin sanitized. For example, math in the section name is removed before creating the filename, so the rest of the filename must be sufficiently unique to avoid name collisions.

⚠ Unique filename!

\FilenameLimit
Default: 80

\FilenameLimit: The maximum length of the filenames generated by lwarp. ".html" is added to this length. Redefine with \renewcommand.

FootnoteDepth (*Ctr*)
Default: 3

FootnoteDepth: Determines where to place pending footnotes. 3 places footnotes before each break down to the \subsubsection level. 1 places footnotes before each \section break. Any pending footnotes are also placed at the bottom of each page before each file break.

FixSmallCaps (*bool*)
Default: false

FixSmallCaps: Set true if SMALL CAPS are rendering in all caps ("SMALL

CAPS"). May be required for some fonts (erewhon, utopia, fbb, et al.), and packages such as embrac.

HTMLDebugComments (*bool*)
Default: false

HTMLDebugComments: Set true to generate HTML comments, such as which section or is being opened or closed.

\abstractname
Default: Abstract

\abstractname: The name of the abstract. This may also be over-written by the babel package. Defaults to "Abstract". Redefine with \renewcommand.

\IndexPageSeparator
Default: ", "

\IndexPageSeparator: Index page list separator. Adjust to match index style file. If using gindex, this is set automatically to gindex's \indexpagessep.

\IndexRangeSeparator
Default: "--"

\IndexRangeSeparator: Index page range separator. Adjust to match index style file. If using gindex, this is set automatically to gindex's \indexrangesep.

Placed before \begin{document}, or before any sectioning command which causes a file break:

\CSSFilename
Default: lwarp.css

\CSSFilename: {⟨*filename.css*⟩} Sets the css file to use for the following files. May be changed before each each sectioning command which would cause a file split.

The css styles of the web pages are set by the \CSSFilename command. If \CSSFilename is not used, a default plain style is used to mimic printed LaTeX output. lwarp_sagebrush.css is a semi-fancy colored style as shown in this tutorial. Change it to lwarp_formal.css for a more formal look, or comment out the \CSSFilename command to see the default. \CSSFilename may be used before each file break to set the css for individual pagess of the website.

\MathJaxFilename
Default: lwarp_mathjax.txt

\MathJaxFilename: {⟨*filename*⟩} Sets the MathJax script file to use for the following files. May be changed before each each sectioning command which would cause a file split.

The MathJax script file is copied into the head of each HTML file. This may be used to point to a local repository, add extensions, or change the script somewhere in the middle of the document. \MathJaxFilename may be used before each file break to set the script file for individual pages of the website.

\HTMLLanguage
Default: en-US

\HTMLLanguage: {⟨*langauge*⟩} The HTML file's HTML lang meta tag. Defaults to en-US.

\HTMLTitle
Default: \thetitle

\HTMLTitle: {⟨*title*⟩} Overrides \title for the HTML header's meta title. Defaults to \thetitle, which is set by \title, or empty otherwise. Unlike the author, \thetitle is set by \title even if not using the titling package.

\HTMLTitleBeforeSection
Default: \HTMLTitleBeforeSection

\HTMLTitleBeforeSection: Sets subpage <title> tags to show the website title followed by the section name.

\HTMLTitleAfterSection

\HTMLTitleAfterSection: Sets subpage <title> tags to show the section name followed by the website title.

custom <title>

To customize subpage <title>s, redefine \theHTMLTitleSection, which defaults to:

```
\def\theHTMLTitleSection{%
  \theHTMLTitle\theHTMLTitleSeparator\theHTMLSection%
}
```

\HTMLAuthor
Default: \theauthor

\HTMLAuthor: {⟨*author*⟩} The HTML header's meta author. Defaults to \theauthor, which is set by \author if using the titling package, but is empty otherwise. There are several ways to represent the author and affiliations, especially if using the authblk package, most of which do not result in a sensible \theauthor, so \HTMLAuthor is useful to create a list of authors without their affiliations.

\HTMLDescription
Default: <empty>

\HTMLDescription: {⟨*description*⟩} Sets the HTML description tag for the following files. May be changed before each each sectioning command which would cause a file split.

\HTMLPageTop
Default: <empty>

\HTMLPageTop: {⟨*contents*⟩} A user-definable custom action applied to the top of pages other than the home page. Useful for logos, etc. Defaults empty. \LinkHome may be used to place a link back to the homepage, as well as \LinkPrevious and \LinkNext. Ignored in print output.

\HTMLPageBottom
Default: <empty>

\HTMLPageBottom: {⟨*contents*⟩} A user-definable custom action applied to the bottom of pages other than the home page. Useful for authors, copyright notices, contact information, etc. Defaults empty. \LinkHome may be used to place a link back to the homepage, as well as \LinkPrevious and \LinkNext. Ignored in print output.

\LinkHome

\LinkHome: Creates a link to the home page. Usually used in \HTMLPageTop and related.

\LinkPrevious

\LinkPrevious: Creates a link to the previous HTML page, unless already at the home page. Usually used in \HTMLPageTop and related.

\LinkNext

\LinkNext: Creates a link to the next HTML page, unless already at the end. Usually used in \HTMLPageTop and related.

Placed in the home page before the first sectioning command which causes a file break:

\tableofcontents
⚠ TOC on the homepage!

\tableofcontents: Used to place a table of contents on the home page. This command must be used before the first file split, so that a way is available to navigate to other files from the homepage.

Links to each chapter/section are provided, as selected by tocdepth.

Placed in the document wherever necessary:

\ImageAltText
Default: image

\ImageAltText: Redefine with \renewcommand. \includegraphics and other images are assigned an HTML alt tag according to \ImageAltText along with \AltTextOpen and \AltTextClose. This text is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "image", and it may be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following \includegraphics and other images.

\ThisAltText

\ThisAltText: {⟨*text*⟩} \ThisAltText can be used to assign an HTML alt text attribute to the next image generated by a lateximage, picture, tikzpicture, or any other similar environment which generates an

image, or the next SVG math expression. This tag is cleared after use. The tag is also cleared after each MathJax expression, in case the user changes between SVG math and MathJax.

`\ThisAltText` also may be used to add an HTML `title` to a reference or hyperlink, such as a `\ref`, `\cref`, `\href`, `\url`, `\hyperref`, or `\hyperlink`. In each case, the alternative text is cleared after use.

`\MathImageAltText`
Default: `math image`

`\MathImageAltText`: Redefine with `\renewcommand`. When creating an SVG math image, its HTML `alt` tag may be set to the math expression, which may be hashed for image reuse. In the case of `\ensuremath` or after `\inlinemathother`, where the contents require a unique image for each instance of the same expression, the `alt` tag is set to `\MathImageAltText`, along with `\AltTextOpen` and `\AltTextClose`, and the image is not reused.

This `alt` expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "`math image`", and it may be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following SVG math images.

`\PackageDiagramAltText`
Default: `diagram`

`\PackageDiagramAltText`: Redefine with `\renewcommand`. For many packages, the output is placed inside a `lateximage` with an HTML `alt` tag set to the package name followed by `\PackageDiagramAltText`. For example:

```
(-xy- diagram)
```

This expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "diagram", and may it be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following package diagrams.

`\AltTextOpen`
Default: `(`

`\AltTextOpen`: Redefine with `\renewcommand`.

`\AltTextClose`
Default: `)`

`\AltTextClose`: Redefine with `\renewcommand`. HTML `alt` text is enclosed by the macros `\AltTextOpen` and `\AltTextClose`, which default to an opening and closing parenthesis.

`\HTMLnewcolumntype`

`\HTMLnewcolumntype`: `\newcolumntype` may not always work with lwarp for HTML output, since it often involves TEX boxes and fills. To provide a simplified column type for HTML, add `\HTMLnewcolumntype` in addition.

`warpprint` (*env.*)

`warpprint`: An environment which is only used while generating print output. Place inside anything which does not apply to HTML and which may cause problems with lwarp. If lwarp knows about and emulates or supports a package then its related macros, lengths, counters, etc. probably won't have to be placed inside a `warpprint` environment, but unknown packages may cause problems which may be isolated from lwarp using this environment.

⚠ *Do not place anything else on the same line as* `\end{warpprint}`. Also do not nest `warpprint` inside itself.

`warpHTML` (*env.*)

`warpHTML`: An environment which is only included while generating HTML output. This is useful for website logos and other items which have no purpose in printed output.

⚠ *Do not place anything else on the same line as* `\end{warpHTML}`. Also do not nest `warpHTML` inside itself.

`\warpprintonly` **`\warpprintonly:`** {⟨*contents*⟩} A macro version of the `warpprint` environment.

`\warpHTMLonly` **`\warpHTMLonly:`** {⟨*contents*⟩} A macro version of the `warpHTML` environment.

7.6.1 Example HTML file naming

Examples of ways to name or number HTML files:

Numbered HTML nodes:

Example: Homepage `index.html`, and `node-1`, `node-2`. [13]

```
\usepackage[
    HomeHTMLFilename=index,
    HTMLFilename={node-}
]{lwarp}
\boolfalse{FileSectionNames}
```

Named HTML sections, no prefix:

Example: `index.html`, and `About.html`, `Products.html`

```
\usepackage[
    HomeHTMLFilename=index,
    HTMLFilename={}
]{lwarp}
\booltrue{FileSectionNames}
```

Named HTML sections, with prefix:

Example: Homepage `mywebsite.html`, and additional pages such as `mywebsite-About.html`, `mywebsite-Products`, etc.

```
\usepackage[
    HomeHTMLFilename=mywebsite,
    HTMLFilename={mywebsite-}
]{lwarp}
\booltrue{FileSectionNames}
```

7.7 Customizing the CSS

`\CSSFilename` {⟨*filename*⟩}

Default: `lwarp.css`

`\CSSFilename` may be used to choose which `.css` file is used to display each page of the web site. Use `\CSSFilename` before `\begin{document}` to assign the style of the home page. If different parts of the website should have different styles, call `\CSSFilename` again before each section heading which creates a new file. This may be changed numerous times throughout the file, resulting in different HTML pages having different CSS files assigned:

[13] See `\SetHTMLFileNumber` to number in groups by chapter, for example.

```
...
\CSSFilename{myCSS.css}
\chapter{Another Chapter}
...
```

The styles provided by lwarp include:

lwarp.css: A default style if \CSSFilename is not used. This style is comparable to a plain LaTeX document. To set this style, you may use \CSSFilename{lwarp.css}, or no \CSSFilename call at all.

lwarp_formal.css: A formal style with a serif fonts and a traditional look.

lwarp_sagebrush.css: A style with muted colors, gradient backgrounds, additional borders, and rounded corners.

To see each style in use, change the \CSSFilename entry in the tutorial, lwarpmk html again, and then reload the tutorial webpage.

Custom css

A customized style may also be created. For each new project a file called sample_project.css is generated. This may be renamed to <project>.css then used by assigning \CSSFilename{<project>.css}.

⚠ Rename it!

Note that sample_project.css is overwritten whenever lwarp is loaded in print mode. It is therefore important to rename the file to something like <project>.css before using it, so that your own changes are not overwritten.

<project>.css has an entry which loads lwarp.css, and this entry may be changed to load lwarp_formal.css or lwarp_sagebrush.css if desired. Additional changes to the css may be made by making entries later in the <project>.css file.

lwarp.css (*file*)
project.css (*file*)
sample_project.css (*file*)

It is best to make a local project-specific css file such as project.css, containing only things which are different from lwarp.css. The file project.css should refer to lwarp.css as follows:

```
/* ( --- Start of project.css --- ) */
/* ( --- A sample project-specific CSS file for lwarp --- ) */

/* Uncomment one of the following: */
@import url("lwarp.css") ;
/* @import url("lwarp_formal.css") ; */
/* @import url("lwarp_sagebrush.css") ; */

/* Project-specific CSS setting follow here. */
/* . . . */

/* ( --- End of project.css --- ) */
```

Finally use \CSSFilename{<project>.css} in the document to activate the custom css.

7.8 Assigning css classes and styles

HTML CSS classes and styles may be assigned to fragments of the document.

BlockClass (*env.*) [⟨*style*⟩] {⟨*class*⟩}

An entire block of text, including paragraphs, may be assigned a CSS class and optional CSS style using the BlockClass environment. The result is placed inside a . A BlockClass may nest other BlockClasses or \InlineClasses.

\InlineClass (⟨*wp css style*⟩) [⟨*web css style*⟩] {⟨*css class*⟩} {⟨*text*⟩}

A section of text without paragraphs may be assinged a CSS class and optional CSS style using the \InlineClass macro. The result is placed inside a . \InlineClass may be nested, but per the HTML standard it must not contain BlockClass, nor may it contain a paragraph, nor several other objects such as HTML figures. \InlineClass also accepts a second optional parameter, enclosed inside parentheses, which assigns the style while generating output for a word processor, while ignoring the web style.

Nullified versions of BlockClass and \InlineClass are provided for the print version, so they may be used in the document without placing them inside warpHTML or \warpHTMLonly.

7.9 Selecting the operating system

Unix (*Prog*)
Mac OS (*Prog*)
Linux (*Prog*)

lwarp tries to detect which operating system is being used. UNIX / MAC OS / LINUX is the default (collectively referred to as "UNIX" in the configuration files), and MS-WINDOWS is supported as well.

MS-Windows (*Prog*)
Windows (*Prog*)

If MS-WINDOWS is not correctly detected, use the lwarp option OSWindows.

OSWindows (*Opt*)

When detected or specified, the operating-system path separator used by lwarp is modified, and the boolean usingOSWindows is set true. This boolean may be tested by the user for later use.

7.10 Selecting actions for print, HTML, or MATHJAX output

The following environments and macros are used to select actions which only apply to either traditional LaTeX print-formatted PDF generation, or to HTML generation, or to HTML with MATHJAX.

For most of built-in LaTeX and many additional packages there is user-level source code support or emulation, so no special handling will be required. For those cases which lwarp does not handle by itself, the following environments and macros may be used to isolate sections of code for print-only or HTML-only.

These environments are also useful for creating a special version of the titlepage for print and another for HTML.

warpHTML (*env.*)

Anything which is to be done only for HTML5 output is surrounded by a warpHTML environment:

```
\begin{warpHTML}
 ... something to be done only during \HTML\ generation
\end{warpHTML}
```

⚠ \end{warpHTML} Do *not* place anything else on the same line as \end{warpHTML}. The exact phrase is used to mark the end of the environment. Do not nest warpHTML inside itself.

⚠ nesting warpMathJax may be used inside warpHTML.

warpprint (*env.*) Anything which is to be done only for print output is surrounded by a warpprint environment:

```
\begin{warpprint}
 ... something to be done only during traditional \PDF\ generation
\end{warpprint}
```

⚠ \end{warpprint} As above, do not place anything else on the line with \end{warpprint}. Do not nest warpprint inside itself.

⚠ nesting

warpall (*env.*) Anything which is to be done for any output may be surrounded by a warpall environment. Doing so is optional.

```
\begin{warpall}
 ... something to be done during print \PDF\ or \HTML\ output
\end{warpall}
```

⚠ \end{warpall} As above, do not place anything else on the line with \end{warpall}. Do not nest warpall inside itself.

⚠ nesting

Macros are also provided for print-only or HTML-only code:

\warpprintonly {⟨*actions*⟩}

Performs the given actions only when print output is being generated.

\warpHTMLonly {⟨*actions*⟩}

Performs the given actions only when HTML output is being generated.

warpMathJax (*env.*) Anything which is to be done only while using HTML output with MathJaxis surrounded by a warpMathJax environment. Usually, this is \CustomizeMathJax, used to add emulation macros. \end{warpMathJax} must appear on its own line.

⚠ \end{warpMathJax} Do not nest warpMathJax inside itself. warpMathJax may be used inside warpHTML.

⚠ nesting

warpsvg (*env.*) Anything which is to be done only while using print output or HTML output with SVG math is surrounded by a warpsvg environment. \end{warpsvg} must appear on its own line. Do not nest warpsvg inside itself. warpsvg may be used inside warpHTML.

⚠ \end{warpsvg}

⚠ nesting

\LWR@formatted To define macros or environments which behave differently depending on print or HTML output, see section 36.

7.11 Commands to be placed into the warpprint environment

Certain print-related commands should always be placed inside a warpprint environment, or may need other special handling. These are unrelated to HTML output, but are hard to isolate automatically. For example:

- Paragraph formatting: `\parindent` `\parskip`
- Manual page positions such as the textpos package, which is emulated but only in a limited way.
- Anything changing the page counter. lwarp requires that the page counter not be adjusted during HTML output.

Some packages require additional setup commands. Where these packages are emulated for HTML, setup commands may work for the emulated HTML output as well as for print output. See the details for each package in this document for more information.

Also see section 13: Troubleshooting.

7.12 Title page

In the preamble, place an additional block of code to set the following:

```
\title{Document Title} % One line only
\author{Author One\affiliation{Affiliation One} \and
    Author Two\affiliation{Affiliation Two} }
\date{Optional date}
```

The title is used in the meta tags in the HTML files, unless overridden by `\HTMLTitle`, and the rest are used in `\maketitle`. To use a `\subtitle` or `\published` field, see section 69.8.

`\maketitle` Use `\maketitle` just after the `\begin{document}`, as this will establish the title of the homepage. Optionally, use a `titlepage` environment instead.

`titlepage` (*env.*) The `titlepage` environment may be used to hold a custom title page. The titlepage will be set in a `<div>` class `titlepage`, and `\printtitle`, etc. may be used inside this environment.

`titlingpage` (*env.*) Another form of custom title page, where `\maketitle` is allowed, and additional information may be included as well.

`\title` {⟨*title*⟩}

⚠ HTML corrupted ⚠ newlines Avoid newlines in the `\title`; these will interfere with the file break and CSS detection. Use a `\subtitle` command instead (section 69.8). The title will appear in the document `\maketitle` as a heading `<h1>`. The HTML meta `title` tag will also have this title, unless `\HTMLTitle` is used to set the meta title to something else instead.

`\author` {⟨*author*⟩}

In `\author`, `\protect` may be needed before some formatting commands. In HTML, the author will appear in a `<div>` of class `author` in the `\maketitle`. If the titling package is used, the author will also appear in a HTML meta tag, but `\HTMLAuthor` may be necessary to create a plain list of names if `\author` had affiliations added. `\affiliation` is a new addition to lwarp.

`\date` {⟨*date*⟩}

\date works as expected. In HTML, this will appear in a class titledate.

\thanks {⟨*text*⟩}

\thanks are allowed in the titlepage fields, and will be rendered as HTML notes at the bottom of the title page.

7.13 HTML page meta descriptions

\HTMLDescription {⟨*A description of the web page.*⟩}
Default: (none)

limitations Each page of HTML output should have its own HTML meta description, which usually shows up in web search results, is limited to around 150 characters in length, and should not include the ASCII double quote character (").

placement Use \HTMLDescription just before \begin{document} to set the description of the home page, and also just before each sectioning command such as \chapter or \section where a new file will be generated, depending on FileDepth. For example, if FileDepth is 1, use \HTMLDescription just before each \section command, and that description will be placed inside the HTML page for that \section. The same descrition will be used for all following HTML files as well, until reset by a new \HTMLDescription. It is best to use a unique description for each HTML file.

disabling To disable the generation of HTML description meta tags, use:

```
\HTMLDescription{}
```

7.14 HTML homepage meta title

\HTMLTitle {⟨*title*⟩}
Default: \HTMLtitle{\thetitle}

Sets the contents of the web page <meta name="title"> element. May be set empty to cancel the meta title tag.

See section 7.6 for \HTMLTitleBeforeSection and \HTMLTitleAfterSection, used to set the title for HTML subpages.

7.15 HTML page meta author

\HTMLAuthor {⟨*author*⟩}
Default: \HTMLAuthor{\theauthor}

Sets the contents of the web page <meta name="author"> element. May be set empty to cancel the meta author tag.

\author may be used to create a list of authors and their affiliations, in several formats if using authblk, and these may not successfully parse properly into a sensible list for \theauthor. \HTMLAuthor may be used to set the meta tag to a simple list of names.

8 Special cases and limitations

Some commonly-used LaTeX expressions should be modified as follows to allow for a smooth conversion to both HTML and print-formatted outputs.

Need help?

See the General Index for "how-to", and the Troubleshooting Index if something doesn't work. A Troubleshooting section is also available. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category.

8.1 Things to avoid

In the document, avoid the following:

⚠ options with braces

Package options: Package options may cause problems with lwarp, especially if they include curley braces.

If selecting options with braces in `\usepackage` does not work:

```
\usepackage[font={it,small}]{caption}% does not work
```

... try instead selecting the package options before loading lwarp:

```
\PassOptionsToPackage{font={it,small}}{caption}
...
\usepackage{lwarp}
...
\usepackage{caption}
```

... or try setting package options after the package has been loaded:

```
\usepackage{caption}
\captionsetup{font={it,small}}
```

page counter: Do not adjust the page counter. If doing so is required for the print version, place the adjustment inside a `warpprint` environment.

Custom math environment macros: Do not use expressions such as `\beq` as a replacement for `\begin{equation}`.

Custom macros in section, figure, table names: Custom macros which appear in sectioning commands or float captions then appear in the `.toc`, `.lof`, and `.lot` lists, and should be made robust using `\newrobustcmd` or `\robustify` from etoolbox, xparse, etc.

When setting `FileSectionNames` to `true` to name the HTML files from the section names, the file names are created from sanitized versions of the chapter or section names, but the section names must be plain text or something which expands into plain text. Robust macros will not work at the sectioning level which is used for file names, but a robust macro or other complicated name may be used for the manditory argument of `\chapter`, `\section`, etc., if a plain-text version is also included in the optional argument:

```
\chapter[Plain Name]{\ARobustMacro{Fancy Name}}
```

8.1.1 Invalid HTML

Additionally, some objects are valid LaTeX, but invalid HTML. An example is a tabular inside `\textbf`, since HTML does not allow a table inside a span. lwarp

will create the table, and the browser may support it, but the result is technically invalid.

8.2 Formatting

8.2.1 Text formatting

⚠ \bfseries, etc. `\textbf`, etc. are supported, but `\bfseries`, etc. work only in some situations.

⚠ HTML special chars &, <, and > have special meanings in HTML. If `\&`, `\textless`, and `\textgreater` are used, proper HTML entities will be used, but there may be HTML parsing problems if these special characters occur unescaped in program listings or other verbatim text.

program listings For program listings, the listings package is supported, and its `literate` option is used to convert &, <, and > to proper HTML entities.

verbatim The various `verbatim`-related environments do not convert &, <, and >, so care must be taken to avoid accidentally including valid HTML code inside these environments. Adding a space on either side may be sufficient.

8.2.2 Small caps

`FixSmallCaps` (*bool*) Some fonts, such as erewhon, utopia, or fbb, and some packages such as embrac, copy/paste "Small Caps" as all caps ("SMALL CAPS"), which lwarp then reads as all caps, so the text is printed in all caps. If small caps are being rendered as all caps, set:

```
\booltrue{FixSmallCaps}
```

⚠ CJK fonts Some CJK fonts may not work if `FixSmallCaps` is set `true`.

8.2.3 Horizontal and vertical space and rules

`\hspace` `\hspace` is converted to an inline HTML span of the given width, except that `0` width is ignored, a width of `.16667em` is converted to an HTML thin breakable space (`U+2009`), and a `\fill` is converted to a `\qquad`.

`\vspace` `\vspace` is ignored for HTML.

`~` `\,` ~ and `\,` are converted to HTML entities.

`\kern` `\hskip` `\kern` and `\hskip` are entered into the HTML PDF output as-is, then interpreted by *pdftotext*, and thus usually appear as a single space.

`\rule` `\rule` is converted to an HTML rule of the same dimensions, of the currently selected text color.

`\hrule` `\vrule` Both `\hrule` and `\vrule` are ignored for HTML. To create a horizontal dividing rule across the page, use `\hrulefill` in its own paragraph.

`\hrulefill` `\hrulefill` usually creates a one-inch rule, similar to a "fill in the blank". If it

is used at the start of a new paragraph, it creates a with a thin horizontal border across the page, as would often be done with \hrule.

8.2.4 Text alignment

Use the environments center, flushright, flushleft instead of the macros \centering, \raggedright, \raggedleft.

⚠ figure & table alignment

\centering, etc. are honored in a figure or table if they are the first command inside the float:

```
\begin{table*}
\centering
\caption{A Table}
. . .
```

8.2.5 Accents

Native LaTeX accents such as \" will work, but many more kinds of accents are available when using Unicode-aware XeLaTeX and LuaLaTeX. If using accents in section names which will become file names, it is recommended to use the LaTeX accents such as \" and \v instead of Unicode accents. The LaTeX accents will have the accents stripped when creating the filenames, whereas the Unicode accents will appear in the file names, which may cause issues with some operating systems.

8.2.6 textcomp package

textcomp (*Pkg*) Some textcomp symbols do not have Unicode equivalents, and thus are not supported.

⚠ missing symbols Many textcomp symbols are not supported by many system / browser fonts. In the css try referencing fonts which are more complete, but expect to see gaps in coverage.

8.2.7 Superscripts and other non-math uses of math mode

Use ^{x} instead of ^{x}

8.2.8 Empty \item followed by a new line of text or a nested list:

lists Use a trailing backslash: \item[label] \

8.2.9 Filenames and URLs in lists or footnotes

filename underscore Escape underscores in the filenames:

```
\item[\href{file\_name.pdf}{text}]
```

8.2.10 relsize package

relsize (*Pkg*) For HTML, only the inline macros are supported: \textlarger, \textsmaller, and \textscale. Each becomes an inline span of a modified font-size.

\relsize, \larger, \smaller, and \relscale are ignored.

While creating SVG math for HTML, the original definitions are temporarilty restored, and so should work as expected.

⚠ not small The HTML browser's setting for minumum font size may limit how small the output will be displayed.

8.3 Boxes and minipages

8.3.1 Marginpars

\marginpar [⟨*left*⟩] {⟨*right*⟩} \marginpar may contains paragraphs, but in order to remain inline with the surrounding text lwarp nullifies block-related macros inside the \marginpar. Paragraph breaks are converted to <br /> tags.

\marginparBlock [⟨*left*⟩] {⟨*right*⟩} To include block-related macros, use \marginparBlock, which takes the same arguments but creates a instead of a . A line break will occur in the text where the \marginBlock occurs.

8.3.2 Save Boxes

⚠ HTML corrupted

⚠ boxes TEX boxes are placed inline and do not allow line breaks, so boxes with long contents may overflow the line during HTML conversion. lwarp uses methods which help avoid this problem.

⚠ minipage, \parbox \savebox and related do not (yet) support minipage or \parbox.

8.3.3 Minipages

⚠ inline A line of text with an inline minipage or \parbox will have the minipage or \parbox placed onto its own line, because a paragraph is a block element and cannot be made inline-block.

placement minipages and \parboxes will be placed side-by-side in HTML unless you place a \newline between them.

side-by-side Side-by-side minipages may be separated by \quad, \qquad, \enskip, \hspace, \hfill, or a \rule. When inside a center environment, the result is similar in print and HTML. Paragraph tags are suppressed between side-by-side minipages and these spacing commands, but not at the start or end of the paragraph.

⚠ minipage in a span There is limited support for minipages inside an HTML . An HTML cannot appear inside a . While in a , minipages, and \parboxes, and any enclosed lists have limited HTML tags, resulting in an "inline" format, without markup except for HTML breaks. Use \newline or \par for an HTML break.

⚠ minipage size — When using `minipage`, `\parbox`, and `fminipage`, a virtual 6×9 inch text area is used for `\linewidth`, `\textwidth`, and `\textheight`, both for sizing the minipage, and also for its contents.

if width is `\linewidth` — If a `minipage` or `\parbox` is assigned a width of exactly `\linewidth`, in HTML it is automatically given no HTML width, thus allowed to fill the line as needed, similar to how it appears in print output.

full-width if HTML — A new macro `\minipagefullwidth` requests that, during HTML output, the next single `minipage` or `\parbox` be generated without an HTML `width` attribute, allowing it to be the full width of the display rather than the declared print-output width. This may be useful where the printed version's width makes no sense in HTML.

⚠ `tabular`, `multicols` — Inside a `tabular` or `multicols` environment, where the width depends on the browser window, `\minipagefullwidth` is effectively used by default for every `minipage` or `\parbox` inside the environment. `\UseMinipageWidths` may be used to tell lwarp to honor the specified widths of all following `minipages` and `\parboxes` until the end of the local scope, and `\IgnoreMinipageWidths` may be used to tell lwarp to ignore the specified widths.

`\UseMinipageWidths`

`\IgnoreMinipageWidths`

⚠ multicol — Inside a `multicols`, `\linewidth` is divided by the specified number of columns.

⚠ text alignment — Nested `minipages` adopt their parent's text alignment in HTML, whereas in regular LaTeX PDF output they do not. Use a `flushleft` or similar environment in the child `minipage` to force a text alignment.

8.3.4 Side-by-side minipages

Place side-by-side minipages inside a `center` environment, with horizontal space between them, such as `\quad`, `\qquad`, `\hspace`, or `\hfill`. The result is similar in print and HTML. Do not use space commands at the start or end of the line.

8.3.5 Framed minipages and other environments

`\fbox` can only be used around inline `<span>` items during HTML output, but HTML cannot place a block element such as a `<div>` for a minipage or a list inside of a `<span>`. Several options are provided for framing an object, depending on which kind of object and which packages are loaded:

`\fbox`

`\fboxBlock`

`fminipage` (*env.*)

For a framed object, options include:

To remove the frame in HTML output: Place the `\fbox` command and its closing brace inside `warpprint` environments. This will nullify the frame for HTML output.

For inline text:

To frame the contents inline with some formatting losses in HTML: This is the default action of `\fbox` when enclosing a minipage. During HTML output, `\fbox` nullifies the HTML tags for `minipage`, `\parbox`, and lists. The contents are included as inline text inside the `\fbox`'s `<span>` of class `framebox`. For lists, line breaks are converted to HTML breaks. The result is a plain-text inline version of the contents, framed inline with the surrounding text, but lacking any extra HTML markup.

For inline `minipage` and lists:

To frame the contents on their own line with improved formatting in HTML: A new command `\fboxBlock` is included, intended to be a direct replacement

for \fbox for cases where the \fbox surrounds a minipage, table, or list. For print output, this behaves as \fbox. For HTML output, the contents are placed inside an HTML with the class framed, resulting in the contents being placed on their own line with a frame surrounding them. The contents preserve their HTML formatting, so lists and minipages look nicer, and valid HTML is created for a tabular. While an \fbox containing a tabular is valid LaTeX code, the result in HTML is problematic since a table is a not a , so use \fboxBlock around a tabular, or else place the tabular inside a minipage, or use fminipage, described next. Also see below regarding the "Misplaced alignment tab character &." error.

For display tabular, minipages, and lists:

To create a framed minipage in both print and HTML: A new environment fminipage is included. For print output, this is identical to minipage, except that it is also framed. For HTML output, this forms a of class framed, the contents preserve their HTML formatting, and valid HTML is created for a tabular. Also see section 89 for a new environment fcolorminipage. Also see below regarding the "Misplaced alignment tab character &." error.

colored boxes and frames:

To create colored frames and boxes: See section 674 for xcolor's \colorbox and \fcolorbox, and lwarp's additional \colorboxBlock and \fcolorboxBlock.

⚠ Misplaced alignment tab character &

To frame tables or verbatim environments: Place the contents inside a fminipage, or perhaps a \fboxBlock for a tabular. Also, if using \fboxblock with tabular, you will have to use \StartDefiningTabulars before the start of the macro which uses \fboxBlock and the tabular, and \StopDefiningTabulars afterwards. Also see the lwarp documentation for the fancybox package.

To frame equations: See section 259 for the fancybox package.

For fancy framed minipages: See packages boxedminipage, shadow, fancybox, framed, mdframed.

Custom environments: Use a custom environment to create a sidebar, containing a BlockClass environment with custom CSS formatting, and \warpprintonly{\hrule} command:

```
\begin{BlockClass}{frameminipage}% ignored in print output
  % use \CSS\ to format div class framedminipage
\warpprintonly{\hrule} % only appears in print output
Contents
\warpprintonly{\hrule} % only appears in print output
\end{BlockClass}
```

8.3.6 fancybox package

fancybox (*Pkg*)
framed equation example

fancybox's documentation has an example FramedEqn environment which combines math, \Sbox, a minipage, and an \fbox. This combination requires that the entire environment be enclosed inside a lateximage, which is done by adding \lateximage at the very start of FramedEqn's beginning code, and \endlateximage at the very end of the ending code. Unfortunately, the HTML alt attribute is not used here.

```
\newenvironmentFramedEqn
{
\lateximage% NEW
\setlength{\fboxsep}{15pt}
. . . }{. . .
\[\fbox{\TheSbox}\]
\endlateximage% NEW
}
```

framing alternatives

`\fbox` works with fancybox. Also see lwarp's `\fboxBlock` macro and `fminipage` environment for alternatives to `\fbox` for framing environments.

framed table example

The fancybox documentation's example of a framed table using an `\fbox` containing a `tabular` does not work with lwarp, but the `FramedTable` environment does work if `\fbox` is replaced by `\fboxBlock`. This method does lose some HTML formatting. A better method is to enclose the table's contents inside a `fminipage` environment. The caption may be placed either inside or outside the `fminipage`:

```
\begin{table}
\begin{fminipage}{\linewidth}
\begin{tabular}{lr}
. . .
\end{tabular}
\end{fminipage}
\end{table}
```

⚠ framed verbatim

lwarp does not support the `verbatim` environment inside a span, box, or fancybox's `\Sbox`, but a `verbatim` may be placed inside a `fminipage`. The fancybox documentation's example `FramedVerb` may be defined as:

```
\newenvironment{FramedVerb}[1] % width
{
  \VerbatimEnvironment
  \fminipage{#1}
  \beginVerbatim
}{
  \endVerbatim
  \endfminipage
}
```

framed `\VerbBox`

fancybox's `\VerbBox` may be used inside `\fbox`.

indented alignment

`LVerbatim`, `\LVerbatimInput`, and `\LUseVerbatim` indent with horizontal space which may not line up exactly with what *pdftotext* detects. Some lines may be off slightly in their left edge.

8.3.7 mdframed package

`mdframed` (*Pkg*) support

Most basic functionality is supported, including frame background colors and single-border colors and thickness, title and subtitle background colors and borders and thickness, border radius, and shadow. CSS classes are created for mdframed environments and frame titles.

⚠ loading

When used, lwarp loads mdframed in HTML with `frametheme=none`.

font

For title font, use

```
frametitlefont=\textbf,
```

instead of

```
frametitlefont=\bfseries,
```

where `\textbf` must appear just before the comma and will receive the following text as its argument (since the text happens to be between braces in the mdframed source). Since lwarp does not support `\bfseries` and friends, only one font selection may be made at a time.

theoremtitlefont

`theoremtitlefont` is not supported, since the following text is not in braces in the mdframed source.

ignored options

`userdefinedwidth` and `align` are currently ignored.

css classes

Environments created or encapsulated by mdframed are enclosed in a `<div>` of class `mdframed`, and also class `md<environmentname>` for new environments.

Frame titles are placed in a `<div>` of class |mdframedtitle|. Subtitles are in a `<div>` of class |mdframedsubtitle|, and likewise for subsubtitles.

8.3.8 tcolorbox package

`tcolorbox` (*Pkg*)

tcolorbox is emulated for HTML and MathJax, and supported as-is inside a `lateximage` or SVG math.

What has been tested to work (at least partly) includes:

- `tcolorbox`, `\tcbox`.
- Title, subtitle.
- Upper, lower parts.
- Colors and title fonts.
- Floating objects.
- Some layered box features.
- Counters, labels, references.
- listings, listingsutf8.
- `theorems`: Theorems are supported. `math`, `ams equation`, etc. are not supported. Use a `tcolorbox` with regular math inside it. `\tcboxmath` and `\tcbhighmath` are suppored in SVG math, and emulated in MathJax.

⚠ math

- Fitting features: `\tcboxfit` becomes `\tcbox` in HTML.

⚠ footnotes

- Footnote numbering does not match the printed output.
- MathJax emulation is provided for common macros.

⚠ undefined references

If using cleveref, it may be necessary to name theorems such as:

```
\crefname{tcb@cnt@mytheo}{my theorem}{my theorems}
```

8.4 Section names

If using named HTML files, by selecting `\booltrue{FileSectionNames}`, the generated filenames may be simplified by using `\FilenameSimplify` and `\FilenameNullify`:

`\FilenameSimplify` {⟨*text*⟩}

To remove common short words from the automatically-generated filenames, replacing each with a single hyphen "-", use \FilenameSimplify:

```
\FilenameSimplify*{-in-}
\FilenameSimplify*{A-}
```

The first example removes the word "in" in the middle of a filename, and the second example removes "A" at the start of the filename. The star forces the arguments to be detokenized, which is required for a plain-text comparison. (The unstarred form is used for a token-sensitive comparison, which is seldom required by the user.) After simplfication, repeated hyphen characters will be further simplified to a single hyphen "-". Finally, single hyphens at the start or end of the filename are removed.

\FilenameNullify {⟨*macros*⟩}

⚠ macros in section names

Macro names may appear in the automatically-generated file names. To remove these, create *non-robust* nullified versions of the macros, ensuring that each line ends with a percent character % as shown below. These are placed inside \FilenameNullify, which adds them to the list of macros which are nullfiied during filename generation. Low-level macros such as \begingroup will cause problems when nullfied. Many macros such as \textbf are already nullfied. lwarp also already nullifies built-in symbol and textcomp macros, including if defined by xunicode, but not all xunicode macros. See the definition of \LWR@nullfonts for a complete list.

```
\FilenameNullify{%
  \renewcommand*{\macroname}[1]{#1}%
  \renewcommand*{\anothermacro}{}%
}
```

⚠ duplicate filename

Avoid duplicate file names. Section names at levels which result in HTML file splits must be unique. lwarp will generate an error if a duplicate HTML filename is generated. Use the optional TOC caption entry parameter for formatting. Remember to \protect LaTeX commands which appear in section names and TOC captions.

⚠ math in section names

If using named HTML files, in section names use paren math \(x+y\) instead of dollar math $x+y$. (Dollar math works, but appears in the filename.) Or, use a short name for the TOC entry without the math, or use \texorpdfstring from the hyperref package:

```
\section{Some math \texorpdfstring{$1+2=3$}{three}}
```

8.5 Cross-references

labels
⚠ label characters

Labels with special characters may be a problem. It is best to stick with alphanumeric, hyphen, underscore, and perhaps the colon (if not French).

\nameref
⚠ empty link

\nameref refers to the most recently-used section where the \label was defined. If no section has been defined before the \label, the link will be empty. Index entries also use \nameref and have the same limitation.

8.5.1 Page references

⚠ LaTeX page numbers

The printed page does not translate to the HTML page, so `\pageref` references are converted to parentheses containing `\pagerefPageFor`, which defaults to "see ", followed by a hyperlink to the appropriate object.

Ex:

```
\ref{sec:name} on page \pageref{sec:name}
```

in HTML becomes:

"Sec. 1.23 on page (**see** sec. 1.23)".

`\pagerefPageFor` may be redefined to "page for ", empty, etc. See page 503.

8.5.2 cleveref and varioref packages

`cleveref` (*Pkg*)
`varioref` (*Pkg*)

cleveref and varioref are supported, but printed page numbers do not map to HTML, so a section name or a text phrase are used for `\cpageref` and `\cpagerefrange`. This phrase includes `\cpagerefFor`, which defaults to "for".

⚠ cleveref page numbers

Ex:

```
\cpageref{tab:first,tab:second}
```

in html becomes:

"pages **for** table 4.1 and **for** table 4.2"

See `\cpagerefFor` at page 734 to redefine the message which is printed for page number references.

⚠ varioref types

cleveref changes the behavior of varioref in that the reference type is automatically printed if cleveref is loaded. Lwarp requires cleveref, so the HTML version will always automatically print the reference types even if the print mode does not. The simplest way to make them match is to require the cleveref package for the document.

8.5.3 Hyperlinks, hyperref, and url

`hyperref` (*Pkg*)
`url` (*Pkg*)

lwarp emulates hyperref, including the creation of active hyperlinks, but does not require that hyperref be loaded by the document.

⚠ comments between arguments

Do not place a comment with a `%` character between arguments for `\hyperref`, etc., as it is neutralized for inclusion in HTML URLs.

lwarp can also load url, but url should not be used at the same time as hyperref, since they both define the `\url` command. lwarp does not (yet) attempt to convert url links into hyperlinks during HTML output, nor does the print version of url create hyperlinks.

⚠ backref

When generating HTML, lwarp's emulation of hyperref does not automatically load backref, so backref must be loaded explicitly.

8.5.4 Footnotes, endnotes, and page notes

lwarp uses native LaTeX footnote code, although with its own \box to avoid the LaTeX output routine. The usual functions mostly work as-is.

footnote numbering

To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
— or —
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

MathJax

Also for MathJax, \footnotename is used for a \footnotemark if the actual footnote number is not known. To redefine it, provide it before loading lwarp:

```
\providecommand{\footnotename}{something}
\usepackage{lwarp}
```

Similar for sidenotes. For endnotes:

```
\def\endnotename{something}% \def allows name to start with
"end"
```

For the pagenote package, there is no \pagenotename to define, since there is no \pagenotemark command.

footmisc

The footmisc stable option is emulated by lwarp.

⚠ sectioning commands

When using footnotes in sectioning commands, to generate consistent results between print and HTML, use the footmisc package with the stable option, provide a short TOC entry, and \protect the \footnote:

```
\usepackage[stable]{footmisc}
. . .
\subsection[Subsection Name]
    {Subsection Name\protect\footnote{A footnote.}}
```

memoir with footmisc
⚠ memoir

If using memoir class, with which lwarp preloads footmisc, the stable option must be declared before lwarp is loaded:

```
\PassOptionsToPackage{stable}{footmisc}
\usepackage{lwarp}
. . .
```

Do not use a starred sectioning command. As an alternative, it may be possible to adjust \secnumdepth instead.

fancybox, fancyvrb
⚠ \VerbatimFootnotes
⚠ sectioning or displaymath

If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

At present there is a bug such that paragraph closing tags are not present in footnotes when `\VerbatimFootnotes` are selected. The browser usually compensates.

pfnote
⚠ pfnote numbers

While emulating pfnote, lwarp is not able to reset HTML footnote numbers per page number to match the printed version, as HTML has no concept of page numbers. lwarp therefore uses continuous footnote numbering even for pfnote.

bigfoot, manyfoot
⚠ verbatim

Verbatim footnotes are not yet supported.

If using the bigfoot package, and possibly also manyfoot, problems may occur with counter allocation because lwarp uses many counters, and there is a difference in how counters numbered 256 and up are handled in PDF LaTeX. With bigfoot this has been known to show up as an error related to one footnote insert being forbidden inside another. Another problem showed up as a input stack error, and which of these problems occurred depended on how many counters were allocated.

As a possible solution, try creating several new counters before defining bigfoot or manyfoot footnotes, hoping to shift the problematic counter above the 256 threshold. It may instead be necessary to use XeLaTeX or LuaLaTeX instead of PDF LaTeX.

8.5.5 xr, xr-hyper, and xcite packages

See section 5.17.

8.6 Front and back matter

8.6.1 Custom classes with multiple authors and affiliations

Some classes allow multiple authors and affiliations. Often it is possible to emulate these using a standard class along with authblk:

```
%\documentclass{customclass} % for print document
\documentclass{article} % for html document

\usepackage{lwarp}
\begin{warpHTML}
\usepackage{authblk}
\let\affiliation\affil % maybe required
\end{warpHTML}
```

8.6.2 Starred chapters and sections

HTML page and TOC

The following describes `\ForceHTMLPage` and `\ForceHTMLTOC`, which may be used for endnotes, glossaries, tocbibind, bibliographies, and the index. See the following sections where applicable. Continue here if interested in the reason for adding these commands to lwarp.

Some packages use \chapter* or \section* to introduce reference material such as notes or lists, often to be placed in the back matter of a book. These starred sections are placed inline instead of on their own HTML pages, and they are not given TOC entries.

lwarp provides a method to cause a starred section to be on its own HTML page, subject to FileDepth, and also a method to cause the starred section to have its own TOC entry during HTML output.

\ForceHTMLPage

To place a starred section on its own HTML page, use \ForceHTMLPage just before the \chapter* or \section*. lwarp will create a new page for the starred sectional unit.

A starred sectional unit does not have a TOC entry unless one is placed manually. The typical method using \phantomsection and \addcontentsline works for inline text but fails when the new starred section is given its own webpage after the TOC entry is created, or when creating an EPUB where the TOC entry will point to the page before the starred section. If the starred section has its own HTML page but no correct TOC entry pointing to that page, the page will be inaccessible unless some other link is created.

⚠ inaccessible HTML page

\ForceHTMLTOC

To automatically force the HTML version of the document to have a TOC entry for a starred section, use \ForceHTMLTOC just before the \chapter* or \section*, and place \phantomsection and \addcontentsline inside a warpprint environment.

For print output, \ForceHTMLTOC and \ForceHTMLPage have no effect.

8.6.3 abstract package

abstract (*Pkg*)

⚠ missing TOC

If using the number option with file splits, be sure to place the table of contents before the abstract. The number option causes a section break which may cause a file split, which would put a table of contents out of the home page if it is after the abstract.

8.6.4 titling and authblk

titling (*Pkg*)
authblk (*Pkg*)
package support
⚠ load order
\published and \subtitle

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 69.8.

8.6.5 tocloft package

titles (*Opt*) [tocloft]
tocloft (*Pkg*)
tocloft (*Pkg*)
⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its titles option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

8.6.6 appendix package

appendix (*Pkg*)

⚠ incorrect TOC link

During HTML conversion, the option `toc` without the option `page` results in a TOC link to whichever section was before the `appendices` environment. It is recommended to use both `toc` and also `page` at the same time.

8.6.7 pagenote package

pagenote (*Pkg*)

pagenote works as-is, but the `page` option is disabled.

⚠ labels

Note that labels in page notes do not appear as expected, even in the print version.

8.6.8 endnotes package

endnotes (*Pkg*)

table of contents

To place the endnotes in the TOC, use:

```
\usepackage{endnotes}
\appto\enoteheading{\addcontentsline{toc}{section}{\notesname}}
\renewcommand*{\notesname}{Endnotes} % optional
```

HTML page

To additionally have the endnotes on their own HTML page, if `FileDepth` allows:

```
\ForceHTMLPage
\theendnotes
```

⚠ `\endnotemark` numbering

If using MathJax, see section 8.5.4 regarding the use of `\endnotemark` and `\endnotetext`.

8.6.9 *BibTeX*

`\etalchar`

Displays a superscript "+" to indicate "and others".

⚠ Modify *.bib

When enough authors are cited for a source, BibTeX may use the `\etalchar` command to display a math superscript with a + character to indicate "and others". Without modification, this will result in an "Improper `\prevdepth`" error. At present, lwarp requires that `\etalchar` be replaced by a text superscript. To do so, add to the start of the `.bib` file the following:

```
@PREAMBLE{"\let\etalchar\relax \newcommand{\etalchar}[1]{\textsuperscript{#1}}"}
```

8.6.10 xcite package

See section 5.17.

8.6.11 gloss package

gloss (*Pkg*)

⚠ compiling

To process the HTML glossary:

```
bibtex <projectname>_html.gls
```

8.6.12 glossaries package

glossaries (*Pkg*)
processing glossaries
GlossaryCmd (*Opt*)
Default: makeglossaries
printglossary (*Opt*) [lwarpmk]
htmlglossary (*Opt*) [lwarpmk]

lwarpmk has the commands **lwarpmk printglossary** and **lwarpmk htmlglossary**, which process the glossaries created by the glossaries package using that package's *makeglossaries* program.

The shell command to execute is set by the lwarp option GlossaryCmd, which defaults to **makeglossaries**. The print or HTML glossary filename is appended to this command.

⚠ *makeglossaries* not found

In some situations it may be required to modify the default command, such as to add the **perl** command in front:

```
\usepackage[
   GlossaryCmd={perl makeglossaries},
] {lwarp}
```

xindy language

To set the language to use for processing glossaries with *xindy*:

```
\usepackage[
   GlossaryCmd={makeglossaries -L english},
] {lwarp}
```

Other options for *makeglossaries* may be set as well.

placement and TOC options

The glossaries may be placed in a numbered or unnumbered section, given a TOC entry, and placed inline or on their own HTML page:

Numbered section, on its own HTML page:

```
\usepackage[xindy,toc,numberedsection=nolabel]{glossaries}
...
\printglossaries
```

Unnumbered section, inline with the current HTML page:

```
\usepackage[xindy,toc]{glossaries}
...
\printglossaries
```

Unnumbered section, on its own HTML page:

```
\usepackage[xindy,toc]{glossaries}
...
\ForceHTMLPage
\printglossaries
```

⚠ glossary style

The default style=item option for glossaries conflicts with lwarp, so the style is forced to index instead.

⚠ number list

The page number list in the printed form would become \namerefs in HTML, which could become a very long string if many items are referenced. For now, the number list is simply turned off.

print/HTML versions

The print and HTML versions of the glossary differ in their internal page numbers. Separate commands for generating print and HTML glossaries are used, even though the page number is currently ignored.

8.6.13 nomencl package

nomencl (*Pkg*) To process the HTML nomenclature:

```
makeindex <project>_html.nlo -s nomencl.ist -o
<project>_html.nls
```

8.6.14 Indexing overview

There are many ways to process indexes for a LaTeX document, including native LaTeX capabilities, a number of packages and classes, the possible availability of shell escape and *latexmk*, and the need to process print and HTML versions. lwarp attempts to provide easy recompilation of indexes along with the rest of the document, but the various indexing options must be set correctly. Numerous examples are given below. Some differ in minor details, so the important parts are highlighted in red, and options are in green.

Once set up properly, the entire document may be recompiled with **`lwarpmk print`** and **`lwarpmk html`**. In some cases, it will also be necessary to compile the indexes with **`lwarpmk printindex`** and **`lwarpmk htmlindex`**. A recompile may then be forced with **`lwarpmk print1`** and **`lwarpmk html1`**.

manual processing The user may continue to process indexes manually or by shell script without the use of *lwarpmk*, but adjustments will be required to process HTML indexes as well. In general, `*.idx` and `*.ind` files will be accompanied by `*_html.idx` and `*_html.ind` files.

custom index style If using a custom indexing style file, see sections 8.6.20 to 8.6.22.

link appearance To control how the index links appear in the HTML output, see the `IndexRef` option in section 7.5, page 106.

source code See section 79 for lwarp's core index and glossary code, section 340 for index, section 571 for splitidx, section 338 for imakeidx, section 628 for tocbibind, and section 695.17 for memoir's indexing patches.

8.6.15 Indexing with makeidx, *makeindex*, *xindy*, *xindex*, gindex

lwarpmk processing The following allow the user to process indexes automatically, or using *lwarpmk*'s commands:

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

makeindex (*Prog*) **For a single index using *makeindex*:**

```
\usepackage[makeindex,latexmk] {lwarp}
```

The usual `.idx` and `.ind` files will be used, along with the new `lwarp.ist` style file. When creating the HTML index, "`_html`" is automatically appended to each of the names.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, see section 8.6.20.

`xindy` (*Prog*) **For a single index using *xindy*:**

```
\usepackage[
    xindy,
    xindyLanguage=english,                  <optional>
    xindyCodepage=utf8,                     <optional>
    latexmk                                 <optional>
]{lwarp}
```

The usual `.idx` and `.ind` files will be used, along with the new `lwarp.xdy` style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, see section 8.6.21.

`xindex` (*Prog*) **For a single index using *xindex*:**

```
\usepackage[
    xindex,
    latexmk                                 <optional>
]{lwarp}
```

The usual `.idx` and `.ind` files will be used.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, see section 8.6.22.

`gindex` (*Pkg*) **For a single index using gindex:**

```
\usepackage[
    makeindex,
    makeindexStyle=gindex.ist,
    . . . or . . .
    makeindexStyle=gindexh.ist,
    latexmk                                 <optional>
]{lwarp}
```

The usual `.idx` and `.ind` files will be used.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, copy `gindex.ist` to a new file, modify, then specify it with `MakeindexStyle` as above. lwarp will automatically adapt to gindex's `\indexpagessep` and `\indexrangesep` settings.

8.6.16 Indexing with index

index (*Prog*)

lwarp is told how to use *makeindex* using the PrintIndexCmd and HTMLIndexCmd options. The file lwarp.ist is specified, which generates index letter heads for print output and also allows special HTML formatting for HTML output.

For multiple indexes using *makeindex* and index:

(Assuming that the second index has file extensions .sist and .sind)

```
\usepackage[
  makeindex, latexmk,
  PrintIndexCmd={
    makeindex -s lwarp.ist <projectname>.idx ;
    makeindex -s lwarp.ist
      -o <projectname>.sind <projectname>.sidx
  },
  HTMLIndexCmd={
    makeindex -s lwarp.ist <projectname>_html.idx ;
    makeindex -s lwarp.ist
      -o <projectname>_html.sind <projectname>_html.sidx
  }
]{lwarp}
\usepackage{index}
. . .
\makeindex
\newindex{secondname}{sidx}{sind}{Second Index}
```

⚠ WINDOWS *For Windows, replace the two ";" characters with "&".*

When creating the HTML index, "_html" is automatically appended to the index filenames.

Use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

If the latexmk option is selected for lwarp, *latexmk* will compile the document but will *not* compile the indexes. **lwarpmk printindex** and **lwarpmk htmlindex** will still be required.

8.6.17 Indexing with splitidx

splitidx (*Prog*)

lwarp is told how to use *splitindex* using the PrintIndexCmd and HTMLIndexCmd options. The file lwarp.ist is specified, which generates index letter heads for print output and also allows special HTML formatting for HTML output.

If the latexmk option is selected for lwarp, *latexmk* will compile the document but will *not* compile the indexes. **lwarpmk printindex** and **lwarpmk htmlindex** will still be required.

⚠ \thepage When using \AtWriteToIndex or \AtNextWriteToIndex, the user must not refer

to \thepage during HTML output, as the concept of a page number is meaningless. Instead, do

```
\addtocounter{LWR@autoindex}{1}
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}
```

where the \index-like action occurs, and then refer to \arabic{LWR@autoindex} instead of \thepage where the reference should occur.

See section 695.17 in the lwarp-patch-memoir package for the \@@wrspindexhyp macro as an example.

For multiple indexes using *makeindex* and splitidx:

```
\usepackage[
  makeindex, latexmk,
  PrintIndexCmd={
    splitindex <projectname> -- -s lwarp.ist
  },
  HTMLIndexCmd={
    splitindex <projectname>_html -- -s lwarp.ist
  }
]{lwarp}
\usepackage{splitidx}
. . .
\makeindex
\newindex[Second Index]{secondname}
```

When creating the HTML index, "_html" is automatically appended to each of the names.

Use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes using *xindy* and splitidx:

```
\usepackage[
  xindy, latexmk,
  PrintIndexCmd={
    splitindex -m xindy <projectname> -- -M lwarp.xdy
      -L english -C utf8                    <optional>
  },
  HTMLIndexCmd={
    splitindex  -m  xindy  <projectname>_html  --  -M
lwarp.xdy
      -L english -C utf8                    <optional>
  }
]{lwarp}
\usepackage{splitidx}
. . .
\makeindex
\newindex[Second Index]{secondname}
```

When creating the HTML index, "_html" is automatically appended to each of the names.

Use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

8.6.18 Indexing with imakeidx

imakeidx (*Prog*)

Due to the number of methods which may be used to process multiple indexes, the options for style file and *xindy* language and codepage must be specified in one of several different ways. These are described in detail later in this section, but are summarized here.

If shell escape is used, imakeidx will automatically compile the indexes by itself. Options specifying a custom style file and *xindy* language and codepage must be specified for each `\makeindex` command using its `options=` option, which must include lwarp's special `lwarp.ist` or `lwarp.xdy` file, or a file based on them. If using a custom indexing style file, see sections 8.6.20 to 8.6.22.

The `splitindex` option is also available of shell escape is used, in which case the splitidx package and *splitindex* program will also be used.

If shell escape is not possible, *latexmk* may be used to automatically compile the indexes. The style, language, and codepage options are specified with lwarp's `makeindexStyle`, `xindyStyle`, `xindyLanguage`, and `xindyCodepage` options. These are passed to *latexmk* by *lwarpmk*'s **lwarpmk printindex** and **lwarpmk htmlindex** commands.

Where shell escape and *latexmk* are not possible, *lwarpmk* may be used to manually compile the indexes. lwarp's `PrintIndexCmd` and `HTMLIndexCmd` options are used.

For a single or multiple indexes using *makeindex* and imakeidx:

The index style `lwarp.ist` is automatically used for HTML output. This file turns on letter headings, so it may be desirable to specify it as an option, in which case it will also be used for print output, which will help match the print and HTML output.

```
\usepackage[makeindex,latexmk] {lwarp}
\usepackage[makeindex]{imakeidx}
. . .
\makeindex[options={-s lwarp.ist}]
\makeindex[name=secondname,options={-s lwarp.ist}]
```

imakeidx will automatically compile the indexes. Shell escape is not required while using *makeindex*. `latexmk` may be specified, and if so it will be used for **lwarpmk print** and **lwarpmk html**, but *imakeidx* will actually create the indexes.

For a single or multiple indexes using *makeindex* and *splitindex* with imakeidx:

The index style `lwarp.ist` is automatically used for HTML output. This file turns on letter headings, so it may be desirable to specify it as an option, in which case it will also be used for print output, which will help match the print and HTML output.

```
\usepackage[makeindex,latexmk] {lwarp}
\usepackage[makeindex,splitindex]{imakeidx}
. . .
\makeindex[options={-s lwarp.ist}]
\makeindex[name=secondname,options={-s lwarp.ist}]
```

⚠ enable shell escape

Shell escape is required while using *splitindex*. For the first compile, use

Enter ⇒ **pdflatex --shell-escape projectname.tex**

Enter ⇒ **pdflatex --enable-write18 projectname.tex** **(MiKTeX)**

or similar with *xelatex* or *lualatex*. lwarp will remember that shell escape was used.

imakeidx will automatically execute *splitindex*, and will also use *makeindex* to compile the indexes.

`latexmk` may be specified, and if so it will be used for **`lwarpmk print`** and **`lwarpmk html`**, but *imakeidx* will actually create the indexes.

For multiple indexes using *xindy* and imakeidx, using shell escape:

Options may be given to imakeidx's `\makeindex` command. The style file `lwarp.xdy` is automatically used for HTML output, and is not necessary for print output since the output will be similar. If language or codepage must be set, they should be specified as options for `\makeindex`, since imakeidx will process the indexes.

```
\usepackage[xindy,latexmk] {lwarp}
\usepackage[xindy,splitindex]{imakeidx}
. . .
\makeindex[
  options={ -M lwarp.xdy -L english -c utf8 }
]
\makeindex[
  name=secondname,
  options={ -M lwarp.xdy -L english -c utf8 }
]
```

⚠ enable shell escape

For the first compile, use

Enter ⇒ **pdflatex --shell-escape projectname.tex**

Enter ⇒ **pdflatex --enable-write18 projectname.tex** **(MiKTeX)**

or similar with *xelatex* or *lualatex*. lwarp will remember that shell escape was used.

imakeidx will automatically execute *splitindex* if selected, and will also use *xindy* to compile the indexes.

If selected, *latexmk* will automatically recompile the entire document as necessary.

For indexes using *xindy* and *imakeidx*, without shell escape, but *with latexmk*:

lwarp's options are used, and are passed to *latexmk*.

```
\usepackage[
  xindy,
  xindyLanguage=english,                    <optional>
  xindyCodepage=utf8,                       <optional>
  latexmk,
]{lwarp}
\usepackage[xindy]{imakeidx}
. . .
\makeindex
\makeindex[name=secondname]
```

latexmk will create the indexes automatically when **`lwarpmk print`** and **`lwarpmk html`** are executed.

For indexes using *xindy* and *imakeidx*, without shell escape, and *without latexmk*:

lwarpmk must be told how to create the indexes:

```
\usepackage[
  xindy,
  PrintIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname.idx
  },
  HTMLIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>_html.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname_html.idx
  }
]{lwarp}
\usepackage[xindy]{imakeidx}
. . .
\makeindex
\makeindex[name=secondname]
```

⚠ WINDOWS *For Windows, replace the two ";" characters with "&".*

`<projectname>` is the `\jobname`: if compiling "`name.tex`", use the filenames `name.idx` and `name_html.idx`.

Use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

For multiple indexes using *xindex* and imakeidx, using shell escape:

xindex, *makeindex*, *imakeidx*, and *splitindex* can all work together:

```
\usepackage[%
  xindex,
  xindexConfig=-imakeidx,
  latexmk
] {lwarp}
\usepackage[makeindex,splitindex]{imakeidx}
. . .
\makeindex[%
  options={ -s lwarp.ist} }
]
\makeindex[
  name=secondname,
  options={ -s lwarp.ist} }
]
```

⚠ enable shell escape

For the first compile, use:

Enter ⇒ **pdflatex --shell-escape projectname.tex**

Enter ⇒ **pdflatex --enable-write18 projectname.tex** **(MiKTeX)**

or similar with *xelatex* or *lualatex*. lwarp will remember if shell escape was used.

xindex will use *imakeidx*, and *imakeidx* will automatically execute *splitindex* if selected.

If selected, *latexmk* will automatically recompile the entire document as necessary.

8.6.19 Indexes with memoir

For a single index with memoir and *makeindex*:

```
\documentclass{memoir}
\usepackage[makeindex,latexmk]{lwarp}
. . .
\makeindex
```

The usual `.idx` and `.ind` files will be used, along with the `lwarp.ist` style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes with memoir and *makeindex*, using *latexmk*:

lwarp's options are used, and are passed to *latexmk*.

```
\documentclass{memoir}
\usepackage[makeindex,latexmk]{lwarp}
. . .
\makeindex
\makeindex[secondname]
```

lwarpmk will use *latexmk* to create the indexes automatically when the user executes **lwarpmk print** and **lwarpmk html**.

For multiple indexes with memoir and *makeindex*, *without* latexmk:

lwarpmk must be told how to create the indexes:

```
\documentclass{memoir}
\usepackage[
  makeindex,
  PrintIndexCmd={
    makeindex -s lwarp.ist <projectname>.idx ;
    makeindex -s lwarp.ist secondname.idx
  },
  HTMLIndexCmd={
    makeindex -s lwarp.ist <projectname>_html.idx ;
    makeindex -s lwarp.ist secondname_html.idx
  }
]{lwarp}
. . .
\makeindex
\makeindex[secondname]
```

⚠ WINDOWS *For Windows, replace the two ";" characters with "&".*

`<projectname>` is the `\jobname`: if compiling "`name.tex`", use the filenames `name.idx` and `name_html.idx`.

Use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

For a single index with memoir and *xindy*:

```
\documentclass{memoir}
\usepackage[
    xindy,
    xindyLanguage=english,                  <optional>
    xindyCodepage=utf8,                     <optional>
    latexmk                                 <optional>
]{lwarp}
. . .
\xindyindex
\makeindex
```

The usual `.idx` and `.ind` files will be used, along with the `lwarp.xdy` style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

For multiple indexes with memoir and *xindy*, using latexmk:

lwarp's options are used, and are passed to *latexmk*.

```
\documentclass{memoir}
\usepackage[
    xindy,
    xindyLanguage=english,                    <optional>
    xindyCodepage=utf8,                       <optional>
    latexmk
]{lwarp}
. . .
\xindyindex
\makeindex
\makeindex[secondname]
```

lwarpmk will use *latexmk* to create the indexes automatically.

For multiple indexes with memoir and *xindy*, *without* latexmk:

lwarpmk must be told how to create the indexes:

```
\documentclass{memoir}
\usepackage[
  xindy,
  PrintIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname.idx
  },
  HTMLIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>_html.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname_html.idx
  }
]{lwarp}
. . .
\xindyindex
\makeindex
\makeindex[secondname]
```

⚠ WINDOWS *For Windows, replace the four ";" characters with "&".*

<projectname> is the \jobname: if compiling "name.tex", use the filenames name.idx and name_html.idx.

Use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

8.6.20 Using a custom *makeindex* style file

makeindex (*Prog*)
lwarp.ist (*file*)

When using *makeindex*, *lwarpmk* uses the file lwarp.ist to process the index. This file is over-written by lwarp whenever a print version of the document is processed.

To use a custom *makeindex* style file:

1. Copy `lwarp.ist` to a new filename such as `projectname.ist`
2. Make changes to `projectname.ist`. Keep the lines which refer to `\hyperindexref`. These lines creates the hyperlinks for the HTML index. During print output `\hyperindexref` becomes a null function.
3. If changing

   ```
   delim_n -and- delim_r
   ```

   in `projectname.ist`, then in the document preamble redefine

   ```
   \IndexPageSeparator -and- \IndexRangeSeparator
   ```

   to match.

`makeindexStyle` (*Opt*)

4. In the document source use the `makeindexStyle` option for lwarp:

   ```
   \usepackage[
     . . . other options . . .
     makeindex,
     makeindexStyle=projectname.ist,
   ]{lwarp}
   ```

   Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.
5. Recompile the print version, which causes lwarp to rewrite the `lwarpmk.conf` configuration file. This tells *lwarpmk* to use the custom `projectname.ist` file instead of `lwarp.ist`.

8.6.21 Using a custom *xindy* style file

`xindy` (*Prog*)
`lwarp.xdy` (*file*)

When using *xindy*, *lwarpmk* uses the file `lwarp.xdy` to process the index. This file is over-written by lwarp whenever a print version of the document is processed.

To use a custom *xindy* style file:

1. Copy `lwarp.xdy` to a new filename such as `projectname.xdy`
2. Make changes to `projectname.xdy`.

   Keep the lines which refer to `\hyperindexref`:

   ```
   (define-attributes (("hyperindexref")))
   (markup-locref :open "\hyperindexref{" :close "}")
   ...
   (markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
   ```

   These lines create the hyperlinks for the HTML index. During print output `\hyperindexref` becomes a null function.

   To create custom styles, refer to the lines for `\textbf` and `\textit`.
3. If changing any of

   ```
   markup-locref-list :sep
   markup-locclass-list :open
   markup-locclass-list :sep
   markup-crossref-layer-list :sep
   markup-range :sep
   ```

in `projectname.xdy`, then in the document preamble redefine

```
\IndexPageSeparator -and- \IndexRangeSeparator
```

to match.

`xindyStyle` (*Opt*)

4. In the document source use the `xindyStyle` option for lwarp:

```
\usepackage[
  . . . other options . . .
  xindy,
  xindyStyle=projectname.xdy,
]{lwarp}
```

Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

5. Recompile the print version, which causes lwarp to rewrite the `lwarpmk.conf` configuration file. This tells *lwarpmk* to use the custom `projectname.xdy` file instead of `lwarp.xdy`.

8.6.22 Using a custom *xindex* style file

`xindex` (*Prog*)

To use a custom *xindex* style file:

⚠ filename

1. Copy `xindex-cfg.lua` to a new filename such as `xindex-projectname.lua`. The filename must start with `xindex-` and end with `.lua`.
2. Make changes to `xindex-projectname.lua`.
3. If changing

```
itemPageDelimiter -and- rangeSymbol
```

in `xindex-projectname.lua`, then in the document preamble redefine

```
\IndexPageSeparator -and- \IndexRangeSeparator
```

to match.

`xindexConfig` (*Opt*)

4. In the document source use the `xindexConfig` option for lwarp:

```
\usepackage[
  . . . other options . . .
  xindex,
  xindexConfig=projectname,   % (without xindex- or .lua)
]{lwarp}
```

Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

5. Recompile the print version, which causes lwarp to rewrite the `lwarpmk.conf` configuration file. This tells *lwarpmk* to use the custom `xindex-projectname.lua` file instead of the default `xindex-cfg.lua`.

8.6.23 Additional indexing limitations

⚠ *xindy* with hyperref

xindy and hyperref may not work well together for print output with "see", "see also", reference ranges, or stylized index references. It may be necessary to turn off hyper-referencing for indexes:

```
\usepackage[hyperindex=false]{hyperref}
```

⚠ empty index

If an HTML index is empty, it may be necessary to add the following before lwarp is loaded:

```
\usepackage{morewrites}
\morewritessetup{allocate=10}
. . .
\usepackage{lwarp}
```

makeindex custom display styles

When using *makeindex*, custom display styles are possible:

```
\begin{warpprint}
\newcommand{\notesstyle}[1]{#1nn}
\end{warpprint}

\begin{warpHTML}
\makeatletter
\newcommand{\notesstyle}[1]{\LWR@doindexentry{#1} notes }
\makeatother
\end{warpHTML}
. . .
A sentence.\index{key|notesstyle}
```

xindy custom display styles

For custom styles with *xindy*, see `lwarp.xdy` for `\textbf` and `\textit` as examples.

8.6.24 Index positions, TOC, tocbibind

placement and TOC options

An index may be placed inline with other HTML text, or on its own HTML page:

makeidx (*Pkg*)

Inline, with a manual TOC entry:

A commonly-used method to introduce an index in a LaTeX document:

```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\printindex
```

makeidx (*Pkg*)

On its own HTML page, with a manual TOC entry:

```
\begin{warpprint}
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\end{warpprint}
\ForceHTMLPage
\ForceHTMLTOC
\printindex
```

tocbibind (*Pkg*)

Inline, with an automatic TOC entry:

The tocbibind package may be used to automatically place an entry in the TOC.

```
\usepackage[nottoc]{tocbibind}
. . .
\cleardoublepage
\phantomsection % to fix print-version index link
\printindex
```

tocbibind (*Pkg*) **On its own HTML page, with an automatic TOC entry:**

```
\usepackage[nottoc]{tocbibind}
. . .
\cleardoublepage
\phantomsection % to fix print-version index link
\ForceHTMLPage
\printindex
```

numindex (*Opt*) [tocbibind]
numbered index section

Use the tocbibind numindex option to generate a numbered index. Without this option, the index heading has no number.

Other packages, such as imakeidx, may also have options for including the index in the Table of Contents.

tocloft (*Pkg*)
⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its `titles` option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

8.7 Math

8.7.1 Math in section names

⚠ math in section names

If using named HTML files, in section names use paren math `\(x+y\)` instead of dollar math `$x+y$`. (Dollar math works, but appears in the filename.) Or, use a short name for the TOC entry without the math, or use `\texorpdfstring` from the hyperref package:

```
\section{Some math \texorpdfstring{$1+2=3$}{three}}
```

8.7.2 Math in custom environments

⚠ math in environments

To create an environment which places its contents inside math, instead of:

```
\newenvironment{mymathenv}{ \( starting math}{ending math \) }
```

use:

```
\NewDocumentEnvironment{mymathenv}{b}
  {
    \inlinemathother
    \( starting math #1 ending math \)
    \inlinemathnormal
  }
  {}
```

or:

```
\usepackage{environ}
\NewEnviron{mymathenv}{
  \inlinemathother
  \( starting math \BODY ending math \)
  \inlinemathnormal
}
```

For display math, use `\[`, `\]`, `\displaymathother`, and `\displaymathnormal`.

8.7.3 Rendering tradeoffs

Math rendering

Math may be rendered as SVG graphics or using the MATHJAX JavaScript display engine.

SVG files

Rendering math as images creates a new SVG file for each expression, except that an MD5 hash is used to combine identical duplicates of the same inline math expression into a single file, which must be converted to SVG only once. Display math is still handled as individual files, since it may contain labels or references which are likely to change.

SVG inline

The SVG images are currently stored separately, but they could be encoded inline directly into the HTML document. This may reduce the number of files and potentially speed loading the images, but slows the display of the rest of the document before the images are loaded.

PNG files

Others LaTeX-to-HTML converters have used PNG files, sometimes pre-scaled for print resolution but displayed on-screen at a scaled down size. This allows high-quality print output at the expense of larger files, but SVG files are the preferred approach for scalable graphics.

MathML

Conversion to MathML might be a better approach, among other things allowing a more compact representation of math than SVG drawings. Problems with MathML include limited browser support and some issues with the fine control of the appearance of the result. Also see section 10 regarding EPUB output with MATHJAX.

8.7.4 SVG option

SVG math option

For SVG math, math is rendered as usual by LaTeX into the initial PDF file using the current font[14], then is captured from the PDF and converted to SVG graphics via a number of utility programs. The SVG format is a scalable-vector web format, so math may be typeset by LaTeX with its fine control and precision, then displayed or printed at any size, depending on (sometimes broken) browser support. An HTML `alt` attribute carries the LaTeX code which generated the math, allowing copy/paste of the LaTeX math expression into other documents.

SVG image font size

For the `lateximage` environment, the size of the math and text used in the SVG image may be adjusted by setting `\LateximageFontSizeName` to a font size name — *without the backslash*, which defaults to:

```
\renewcommand{\LateximageFontSizeName}{normalsize}
```

For inline SVG math, font size is instead controlled by `\LateximageFontScale`, which defaults to:

[14] See section 680 regarding fonts and fractions.

```
\newcommand*{\LateximageFontScale}{.75}
```

svg math copy/paste

For svg math, text copy/paste from the HTML `<alt>` tags lists the equation number or tag for single equations, along with the LaTeX code for the math expression. For $\mathcal{AMS}$ environments with multiple numbers in the same environment, only the first and last is copy/pasted, as a range. No tags are listed inside a starred $\mathcal{AMS}$ environment, although the `\tag` macro will still appear inside the LaTeX math expression.

⚠ svg math size, baseline

svg math sizing and baselines are improved if the graphics or graphicx package is loaded. An almost-invisible marker is placed at either end of the image to assist in cropping and computing the baseline. A warning is issued at the end of the compile if graphics or graphicx are not used.

⚠ svg math in TeX boxes

svg math does not work inside TeX boxes, since a `\newpage` is required before and after each image.

8.7.5 MathJax option

MathJax math option

`MathJax` (*Prog*)

The MathJax (`mathjax.org`) LaTeX-math to HTML converter may be used to display math.

When MathJax is enabled, math is rendered twice:

1. As regular LaTeX PDF output placed inside an HTML comment, allowing equation numbering and cross referencing to be almost entirely under the control of LaTeX, and
2. As detokenized printed LaTeX commands placed directly into the HTML output for interpretation by the MathJax display scripts. An additional script is used to pre-set the equation number format and value according to the current LaTeX values, and the MathJax equation numbering system is ignored in favor of the LaTeX internal system, seamlessly integrating with the rest of the HTML output, including any math appearing in non-MathJax svg output.

8.7.6 MathJax rendering options

⚠ fonts

MathJax v3 may render using CHTML or svg. svg display renders italic characters correctly. To select svg rendering, right-click on some math, and select

Math Settings → Math Renderer → SVG

Wait a moment for the math to rerender.

8.7.7 Customizing MathJax

equation numbering

lwarp detects and adjusts MathJax equation numbering format for article and book style equations as well as amsmath `\numberwithin` for chapters, sections, and subsections. Custom equation number formats may be set as follows, for example:

```
\renewcommand*{\theequation}{\Alph{section}.\arabic{equation}}
\AtBeginDocument{
  \renewcommand*{\theMathJaxsection}{\Alph{section}.}
}
```

⚠ subequation The amsmath subequations environment is supported, but only with \alpha subequation numbering.

global customizations MathJax does not have preexisting support every possible math function. Additional MathJax function definitions may be defined in the preamble. These will be declared at the start of each HTML page, and thus will have a global effect across all HTML pages.

Examples:

```
\begin{warpMathJax}
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arcsinh}{\text{arcsinh}}}
\CustomizeMathJax{\newcommand{\arccosh}{\text{arccosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
\end{warpMathJax}
```

⚠ slow compilation To avoid a slowdown in compile speed, use the warpMathJax environment to prevent its contents from being processed in print or SVG math output. Also, place each new definition inside its own \CustomizeMathJax. A warning to this effect is issued if an overly-long definition is attempted.

lwarp already provides MathJax customizations for some packages.

siunitx When using siunitx, a similar process may be used to add custom units:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\myunit}{\mathrm{WXYZ}}}
\CustomizeMathJax{\newcommand{\umyunit}{\mathrm{\micro\myunit}}}
\end{warpMathJax}
```

advanced control For more advanced control over dynamically creating custom definitions, see as an example the lwarp definition for \DeclarePairedDelimiterX, in section 399, mathtools.

local customizations For customizations local to the current HTML page only, macros may be defined as follows:

```
\begin{warpMathJax}
\( \newcommand{\macroname}{. . . } \)
\( \newcommand{\anothername}{. . . } \)
\end{warpMathJax}
```

To maintain compile speed, use the warpMathJax environment, and use a separate math environment for each definition.

\ifstar For MathJax, use \ifstar instead of \@ifstar:

```
\CustomizeMathJax{
  \def\myname{
    \ifstar\starredaction\unstarredaction
    % (Do not place anything after!)
  } }
```

\ifnextchar

For MathJax, use \ifnextchar instead of \@ifnextchar:

```
\CustomizeMathJax{\def\myname{\ifnextchar X \found\notfound}}
```

"X" may be a single ASCII character, or a hex number inside braces, ex:

```
\CustomizeMathJax{\def\myname{\ifnextchar{0x7B}\found\notfound}}
```

Use "(" or "{0x28}" for a left parenthesis, "{0x7B}" for a left brace, "{0x7D}" for a right brace, or "{0x5C}" for a backslash.

8.7.8 MathJax limitations

MathJax limitations
MathJax (*Prog*)

Limitations when using MathJax include:

⚠ \multicolumn, multirow

- MathJax does not support \multicolumn or multirow. These may be used in text tabulars or SVG math, but in MathJax math arrays they are emulated. \multicolumn only fills a single cell, resulting in a short row. \multirow simply prints its text on the first line.

⚠ footnotes

- Footnotes are emulated when used inside a MathJax expression. For an equation with a single footnote, the correct footnote number is used. For non-equations, \footnotename is used instead, since the actual number cannot be tracked. See section 8.5.4 regarding the use of footnotes with MathJax.

⚠ references

- Inside a MathJax expression, references to equations work within the same HTML web page, but do not work when referring to an equation in a different HTML web page. Outside of a MathJax expression, in the text body, references work as expected.

lateximage

- Math appearing inside a lateximage, and therefore also inside a Ti*k*Z or picture environment, is rendered as SVG math even if MathJaxis used in the rest of the document.

siunitx

- For siunitx, see siunitx package, section 8.7.15.

physics

- For physics, see physics package, section 8.7.17.

tabbing

- A tabbing environment is emulated using an HTML <pre>. While MathJax is enabled inside tabbing, the browser may not correctly render the horizontal alignment of the math and text following after on the same line.

\text

- MathJax includes the *textmacros* extension, which supports various macros which are commonly used inside \text, such as \textbf and text accents. Lwarp supports this extension.

⚠ Unicode

- If using DVI LaTeX or PDF LaTeX, unicode input may not appear correctly in MathJax. Either use XeLaTeX or LuaLaTeX, or replace Unicode special characters such as

  ```
  \text{special character æ}
  ```

  with their special macros, such as

  ```
  \text{special character \ae}
  ```

⚠ other macros and packages

- Many other math-related macros and packages are not directly supported by MathJax, including \ensuremath and occasionally-used macros such as \relax. While using MathJax, lwarp provides emulation for many of these

macros, as well as for footnotes and emulation for dozens of packages (see table 2). In many cases these emulations simply ignore the package in a source-compatible way. Others produce a result which represents the meaning, even if they don't look exact. Look up each package in this document for a description of the limitations of each.

8.7.9 Catcode changes

preamble macros with math

The math shift character $ is not set for HTML output until after the preamble. Macros defined in the preamble which contain $ must be enclosed between \StartDefiningMath and \StopDefiningMath to temporarily change to the HTML meaning of $:

```
\StartDefiningMath
\newcommand{. . . }
\StopDefiningMath
```

As an alternative, use \(and \) instead of $, in which case \StartDefiningMath and \StopDefiningMath are not necessary.

If a package defines macros using $, it may be nessary to use \StartDefiningMath and \StopDefiningMath before and after loading the package.

8.7.10 Complicated inline math objects

\inlinemathnormal
\inlinemathother

changing contents
complicated alt tag
MathJax limitations

An inline math expression is usually converted to a reusable hashed SVG math image, or a MathJax expression. The hash or expression depends on the contents of the math expression. In most cases this math expression is static, such as $x+1$, so the image can be reused for multiples instances of the same expression. In some cases, the math expression includes a counter or other object which may change between uses. Another problem is complicated contents which do not expand well in an alt tag. Yet another problem is math packages which are only partially emulated in MathJax. The macro \inlinemathother may be used before a sequence of dynamic or complicated math expressions, and \inlinemathnormal after. Doing so tells lwarp to use unhashed SVG math images for those particular expressins, even if MathJaxis otherwise in use. See section 44.

8.7.11 Complicated display math objects

\displaymathnormal

By default, or when selecting \displaymathnormal, MathJax math display environments print their contents as text into HTML for MathJax to interpret, and SVG display math environments render their contents as SVG images and use their contents as the alt tag of HTML output. To do so, the contents are loaded into a macro for reuse. In some cases, such as complicated Ti*k*Z pictures, compilation will fail.

\displaymathother
MathJax unsupported
complicated alt tag

When selecting \displaymathother, it is assumed that the contents are more complicated than "pure" math. An example is an elaborate Ti*k*Z picture, which will not render in MathJax and will not make sense as an HTML alt tag. In this mode, MathJax is turned off, math display environments become SVG images, even if MathJax is selected, and the HTML alt tags become simple messages. The contents are internally processed as an environment instead of a macro argument, so complicated objects such as Ti*k*Z pictures are more likely to compile successfully.

8.7.12 Theorems

⚠ cref reference format undefined

If the print version does not use cleveref, place all `\theoremstyle` and `\newtheorem` declarations in the preamble inside `\AtEndPreamble`.[15] For some theorems, it may also be required to add inside `\AtEndPreamble` something such as:

```
\usepackage{etoolbox} % for \ifdef, \AtEndPreamble
\AtEndPreamble{ % if not using cleveref package
  \theoremstyle{definition}
  \newtheorem{dtheorem}{Definition}
  . . .
  \ifdef{\cref}{
    \crefname{Proof}{Proof}{Proofs}
  }{}
}
```

8.7.13 ntheorem package

`ntheorem` (*Pkg*)

⚠ Font control

This conversion is not total. Font control is via CSS, and the custom LaTeX font settings are ignored.

⚠ Equation numbering

ntheorem has a bug with equation numbering in $\mathcal{AMS}$ environments when the option `thref` is used. lwarp does not share this bug, so equations with `\split`, etc, are numbered correctly with lwarp's HTML output, but not with the print output. It is recommended to use cleveref instead of ntheorem's `thref` option.

8.7.14 mathtools package

`mathtools` (*Pkg*)

⚠ equation numbering

`showonlyrefs` is disabled, as it conflicts with cleveref, which is used by lwarp. Equation numbers may not match the print version.

⚠ italic correction

`mathic` is not emulated for HTML.

⚠ MathJax

If using MathJax:

- Recent changes may not yet be updated in the MathJax extension, which is used by lwarp.
- mathtools `disallowspaces` does not work for MathJax. Protect brackets which are not optional arguments, such as:

  ```
  \begin{gathered}{}
  [p]=1 . . .
  \end{gathered}
  ```

- `showonlyrefs` does not work in MathJax, and will result in a difference in equation numbering compared to the print version.
- `alignat` in MathJax requires math mode, but in LaTeX it doesn't. It may be required to use `warpHTML` and `warpprint` to isolate a version for each mode.
- `\DeclarePairedDelimiter` and related must be in the preamble before `\begin{document}`.

[15] lwarp uses cleveref for the HTML conversion, and loads cleveref `\AtEndPreamble`, just before `\AtBeginDocument`. This is also before the `.aux` file is read.

8.7.15 siunitx package

siunitx (*Pkg*) siunitx is well supported by lwarp.

Limitations Some general limitations:

fractions Due to *pdftotext* limitations, fraction output is replaced by symbol output for per-mode and quotient-mode.

\cancel is not currently supported for siunitx v3.

Negative values are not automatically colored.

⚠ tabular Tabular S and s columns are rendered as simple c columns, although key settings will be set. If using scientific notation, table-format, table-align-uncertainty, drop-exponent, etc.. use \tablenum for each cell. This is especially required for

⚠ drop-exponent drop-exponent, without which the value will be shown incorrectly.

⚠ table-auto-round table-auto-round is ignored.

Math rendering Math may be rendered in several ways in the same document:

For math mode with SVG display: The original siunitx code is used while generating the SVG image.

For HTML text mode: lwarp uses siunitx code patched for HTML, and simplified units.

For math expressions while using MathJax: A limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. The result usually looks fine, and otherwise is enough to get the meaning across.

Custom units siunitx allows customized units:

\DeclareSIUnit {⟨*name*⟩} {⟨*definition*⟩}

\DeclareSIUnit declares a version of the unit for the print version. This is also used when the unit is printed in SVG math or a lateximage. It is also used for HTML if an HTML-specific version is not defined with \HTMLDeclareSIUnit.

```
\DeclareSIUnit\myunit{\ensuremath{\text{m}_y}}
```

\HTMLDeclareSIUnit {⟨*name*⟩} {⟨*definition*⟩}

⚠ v3 only! Use this after the print unit has been defined. For siunitx v3, \HTMLDeclareSIUnit declares a simplified version of the unit for HTML, for example if the print-mode unit uses TEX boxes or \ensuremath:

```
\HTMLDeclareSIUnit\myunit{\text{m}\textsubscript{\textit{y}}}
```

It is also possible to provide a custom unit for MathJax:

```
\CustomizeMathJax{\newcommand{\myunit}{\text{m}_y}}
```

Predefined units Most units work as-is with HTML. For the following units, lwarp has already set `\HTMLDeclareSIUnit`: `\celsius`, `\arcminute`, `\arcsecond`, `\elementarycharge`, `\clight`, `\bohr`, `\electronmass`, `\hartree`, `\planckbar`.

⚠ MathJax

Document modifications required for MathJax

⚠ `\sisetup`

- Place `\sisetup` in the preamble before `\begin{document}`. Changes made later may be ignored, especially with MathJax. The MathJax emulation also ignores most macro options.

⚠ complex numbers

- Complex numbers are displayed as entered, ignoring `output-complex-root`.

custom units

- Custom units may be added with `\CustomizeMathJax`. For example, from lwarp-common-mathjax-siunitx:

```
\CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
\CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
```

⚠ unit spacing

- Units work better using ~ between units instead of using periods.

⚠ `\square, \cubic`

- To square or cube compound units, enclose the following compound units in braces:

```
\cubic{\centi\meter}
```

  Single units do not require braces.

- For `\numlist`, the argument is printed as text as-is, so use space between semicolons for improved readability.

⚠ `Missing $ inserted`

- If using `parse-numbers = false`, also use `\num` or `\qty`. siunitx=siunitx>Missing $ inserted.

Also see MathJax option, section 8.7.5.

8.7.16 units and nicefrac packages

`units` (*Pkg*)
`nicefrac` (*Pkg*)

units and nicefrac work with lwarp, but MathJax does not have an extension for units or nicefrac. These packages do work with lwarp's option `svgmath`.

8.7.17 physics package

`physics` (*Pkg*)

physics works as-is for HTML with SVG math.

For MathJax, the MathJax v3 physics extension is used.

8.8 Graphics

`graphics` (*Pkg*)
`graphicx` (*Pkg*)
file extensions

Per table 9, image filenames may be specified either with or without an extension. If an extension is given it will be used as-is, for either print or HTML output. If no extension is given, a list of possible extensions is tried, which depends on whether print or HTML is being generated. This allows a PDF file for print and a SVG file for HTML, for example. If no extension is given, the automatic search will only return lowercase extensions, even if the filename actually has an uppercase

⚠ case sensitive

Table 9: \includegraphics and file names

Print image file	HTML image file	Command to use
image.pdf[a]	image.svg[a]	\includegraphics{image}
image.eps[a]	image.svg[a]	\includegraphics{image}
image.jpg	—[b]	\includegraphics{image}
image.png	—[b]	\includegraphics{image}
image.JPG	—[b]	\includegraphics{image.JPG}[c]
image.PNG	—[b]	\includegraphics{image.PNG}[c]
image.jpg	image.gif	\includegraphics{image}

[a:] Must be a lowercase file extension.
[b:] The same file is used for print and HTML.
[c:] The uppercase extension must be specified.

extension, and lwarp cannot get around this problem, so image file extensions must be lowercase to be seen by the HTML browser with lwarp. For example, name the image file image.pdf instead of image.PDF, but refer to it in the source as image, without an extension. For images which may be used as-is with either print or HTML, such as JPG or PNG, you may use a capitalized extension if it is specified in the source, such as image.JPG.

\includegraphics file formats

For \includegraphics with .pdf or .eps files, the user must provide a .pdf or .eps image file for use in print mode, and also a .svg, .png, or .jpg version of the same image for use in HTML.

```
\includegraphics{filename} % print:.pdf/.eps HTML:.svg, etc.
```

For print output, lwarp will automatically choose the .pdf or .eps format if available, or some other format otherwise. For HTML, one of the other formats is used instead.

If a .pdf or .eps image is referred to with its file extension, the extension will be changed to .svg for HTML:

```
\includegraphics{filename.pdf} % uses .svg in html
\includegraphics{filename.eps} % uses .svg in html
```

pdftocairo (*Prog*)
PDF to SVG

To convert a PDF image to SVG, use the utility *pdftocairo*:

Enter ⇒ **pdftocairo -svg filename.pdf**

lwarpmk pdftosvg (*Prog*)

For a large number of images, use *lwarpmk*:

Enter ⇒ **lwarpmk pdftosvg *.pdf** **(or a list of filenames)**

lwarpmk epstopdf (*Prog*)
epstopdf (*Prog*)
epstopdf package

For EPS images converted to PDF using the package epstopdf, use

Enter ⇒ **lwarpmk pdftosvg *.PDF**

to convert to SVG images.

DVI LaTeX

When using DVI *latex*, it is necessary to convert EPS to PDF and then to SVG:

Enter ⇒ **`lwarpmk epstopdf *.eps`** **(or a list of filenames)**

Enter ⇒ **`lwarpmk pdftosvg *.pdf`** **(or a list of filenames)**

PNG and JPG

For PNG or JPGwhile using *pdflatex*, *lualatex*, or *xelatex*, the same file may be used in both print or HTML versions, and may be used with a file extension, but will also be used without the file extension if it is the only file of its base name.

GIF

GIF files may be used for HTML, but another format must also be provided for print output.

file extension priorities

If a file extension is not used, for HTML the file extension priorities are: SVG, GIF, PNG, then JPG.

duplicate files

⚠ image not displayed

A complication occurs if a file of the same name exists elsewhere in the TeX tree, such as a test image from some LaTeX package. TeX looks in the local document directory before considering the directories specified by `\graphicspath`, but the TeX tree is found as "local", so any file in the tree is found before the directories in `\graphicspath`. To use such an image, it must be copied to the document's directory to be used for HTML, and furthermore must be in the document's base directory instead of an images subdirectory.

⚠ graphics vs. graphicx

⚠ viewport

If using the older graphics syntax, use both optional arguments for `\includegraphics`. A single optional parameter is interpreted as the newer graphicx syntax. Note that viewports are not supported by lwarp — the entire image will be shown.

units

For `\includegraphics`, avoid `px` and % units for width and height, or enclose them inside `warpHTML` environments. For font-proportional image sizes, use `ex` or `em`. For fixed-sized images, use `cm`, `mm`, `in`, `pt`, or `pc`. Use the keys `width=.5\linewidth`, or similar for `\textwidth` or `\textheight` to give fixed-sized images proportional to a 6 by 9 inch text area. Do not use the `scale` option, since it is not well supported by HTML browsers.

options

`\includegraphics` accepts `width` and `height`, `origin`, `rotate` and `scale`, plus new `class` and `alt` keys. (`alt` has recently been incorportated into graphicx itself.)

HTML class

With HTML output, `\includegraphics` accepts an optional `class=xyz` keyval combination, and if this is given then the HTML output will include that class for the image. The class is ignored for print output.

HTML `alt` tags

Likewise, the `\includegraphics` `alt` key adds an HTML `alt` tag to an image, and is ignored for print output. If not assigned, each image is given an `alt` tag according to `\ImageAltText`.

⚠ `scale`

Avoid using the `\includegraphics` `scale` option. Change:

```
\includegraphics[scale=<xx>]{ . . . }
```

to:

```
\includegraphics[width=<yy>\linewidth]{ . . . }
```

`\rotatebox`

`\rotatebox` accepts the optional `origin` key.

⚠ browser support

`\rotatebox`, `\scalebox`, and `\reflectbox` depend on modern browser support. The css3 standard declares that when an object is transformed the whitespace

which they occupied is preserved, unlike LaTeX, so expect some ugly results for scaling and rotating.

8.8.1 tikz package

tikz (*Pkg*)

⚠ displaymath and matrices

If using display math with `tikzpicture` or `\tikz`, along with matrices with the `&` character, the document must be modified as follows:

```
\usepackage{tikz}
\tikzset{every picture/.style={ampersand replacement=\&}}
```

and each instance of `&` in the tikz expression must be replaced with `\&`.

8.8.2 grffile package

grffile (*Pkg*)

⚠ matching PDF and SVG

grffile is supported as-is. File types known to the browser are displayed, and unknown file types are given a link. Each PDF image for print mode should be accompanied by an SVG, PNG, or JPG version for HTML.

8.8.3 color package

color (*Pkg*)

color is superceded by xcolor, and lwarp requires several of the features of xcolor. When color is requested, xcolor is loaded as well.

8.8.4 xcolor package

xcolor (*Pkg*)

`\colorboxBlock` and `\fcolorboxBlock`

`\colorboxBlock` and `\fcolorboxBlock` are provided for increased HTML compatibility, and they are identical to `\colorbox` and `\fcolorbox` in print mode. In HTML mode they place their contents into a `<div>` instead of a `<span>`. These `<div>`s are set to `display: inline-block` so adjacent `\colorboxBlock`s appear side-by-side in HTML, although text is placed before or after each.

Print-mode definitions for `\colorboxBlock` and `\fcolorboxBlock` are created by lwarp's core if xcolor is loaded.

background: none

`\fcolorbox` and `\fcolorboxBlock` allow a background color of `none`, in which case only the frame is drawn, which can be useful for HTML.

color support

Color definitions, models, and mixing are fully supported without any changes required.

colored tables

`\rowcolors` is supported, except that the optional argument is ignored so far.

colored text and boxes

`\textcolor`, `\colorbox`, and `\fcolorbox` are supported.

`\color` and `\pagecolor`

`\color` and `\pagecolor` are ignored. Use CSS or `\textcolor` where possible.

8.8.5 epstopdf package

epstopdf (*Pkg*)

Images with an `.eps` extension will be converted to `.pdf`. The HTML output uses

⚠ convert to .svg

the .svg version, so use

Enter ⇒ **lwarpmk pdftosvg <listofPDFfiles>**

to generate .svg versions.

8.8.6 pstricks package

pstricks (*Pkg*)

All pstricks content should be contained inside a pspicture environment.

⚠ use pspicture

8.8.7 pdftricks package

pdftricks (*Pkg*)

The pdftricks image files <jobname>-fig*.pdf must be converted to .svg, or else a missing file error will occur. The image files must also be converted again whenever they change. To convert the images:

⚠ convert image files

Enter ⇒ **lwarpmk pdftosvg <jobname>-fig*.pdf**

8.8.8 psfrag package

psfrag (*Pkg*)

The psfrags environment is modified to use lateximage to encapsulate the image. Always use a psfrags environment to contain any local \psfrag macros and the associated \includegraphics or \epsfig calls. Outside of a psfrags environment, psfrags adjustments will not be seen by lwarp.

⚠ use psfrags

⚠ Tip: Use a mono-spaced font for the tags in the EPS file.

8.8.9 pstool package

pstool (*Pkg*)

\graphicspath is ignored, and the file directory must be stated.

⚠ path and filename

The filename must not have a file extension.

Use

Enter ⇒ **lwarpmk html**

followed by

Enter ⇒ **lwarpmk limages**

.

8.8.10 asymptote package

asymptote (*Pkg*)

To compile:

```
pdflatex project.tex
asy project-*.asy
pdflatex project.tex

lwarpmk print
asy project-*.asy
lwarpmk print1
lwarpmk print1

lwarpmk html
asy project_html-*.asy
lwarpmk html1
lwarpmk html1
lwarpmk limages
```

8.8.11 overpic package

overpic (*Pkg*)

⚠ scaling

The macros `\overpicfontsize` and `\overpicfontskip` are used during HTML generation. These are sent to `\fontsize` to adjust the font size for scaling differences between the print and HTML versions of the document. Renew these macros before using the `overpic` and `Overpic` environments.

8.8.12 Multimedia packages

multimedia (*Pkg*)

movie15 (*Pkg*)

media9 (*Pkg*)

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for `.mp3` and `.mp4` files.

HTML5 <embed> objects are created for `http` and `ftp` links.

`\href` links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each `addresource=`, as well as each `flashvars` `source=` and `src=`. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside `\warpprintonly` or the `warpprint` environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the `text` option is supported to specify the poster text.

The `width`, `height`, and `totalheight` options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 `\addmediapath` is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each `<audio>` and `<video>` object.

media9 slideshows are not supported.

`\hyperlinkmovie`, `\movieref`, and `\mediabutton` are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with `.../embed/...` instead of `.../v/...`

8.9 Tabbing

The `tabbing` environment works, except that SVG math and `lateximages` do not yet work inside the environment.

⚠ math in `tabbing`

If math is used inside `tabbing`, place `tabbing` inside a `lateximage` environment, which will render the entire environment as a single SVG image.

8.10 Tabular

8.10.1 `tabular` environment

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, siunitx `S` columns, or the packages multirow, longtable, supertabular, or xtab.

Defining macros and environments:

⚠ Misplaced alignment tab character &

- When defining environments or macros which include `tabular` and instances of the & character, it may be necessary to make & active before the environment or macro is defined, then restore & to its default catcode after, using the following commands. These are are ignored in print mode.

```
\StartDefiningTabulars
<define macros or environments using tabular and &
here>
\StopDefiningTabulars
```

⚠ floatrow

This includes before and after defining any macro which used `\ttabbox` from floatrow.

⚠ `tabular` inside another environment

- When creating a new environment which contains a `tabular` environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use `\ResumeTabular` as follows. This is ignored in print mode.

```
\StartDefiningTabulars  %    (&  is  used  in  a
definition)
\newenvironment{outerenvironment}
{
  \tabular{cc}
  left & right \\
}
{
  \TabularMacro\ResumeTabular
  left & right \\
  \endtabular
}
\StopDefiningTabulars
```

For developers:

- To automate the use of `\StartDefiningTabulars` and `\EndDefiningTabulars`, these macros may be embedded inside an HTML environment definition to automatically change the catcode of & before absorbing the arguments. Another environment may be embedded as well.

```
% Does the work after the catcode has been changed:
\newcommand*{\LWR@HTML@subsomename}[2]{%
  . . .
  \otherenvironmentname  [<args>]  {<args>}  %  for
example
}
% Change catcode before absorbing arguments:
\newcommand*{\LWR@HTML@somename{%
  \StartDefiningTabulars
  \LWR@HTML@subsomename
}
% Change catcode again at the end:
\newcommand*{\LWR@HTML@endsomename}{%
  . . .
  \endotherenvironmentname   % for example
  \StopDefiningTabulars
}
% Combine with the existing print definition:
\LWR@formattedenv{somename}
```

Cell contents:

⚠ macro in a table

- Using a custom macro inside a tabular data cell may result in an extra HTML data cell tag, corrupting the HTML table. To avoid this, use `\TabularMacro` just before the macro. This is ignored in print mode.

```
\TabularMacro\somemacro & more row contents \\
```

Column specifiers:

⚠ math

- Due to the way math is gathered for processing, column specifiers such as `>{$}c<{$}` do not work with lwarp. Instead, each cell must specify math mode individually.

`@` and `!`

- Only one each of `@` and `!` is used at each column, and they are used in that order.

`\multirow`

- In `\multirow` cells, the print version may have extra instances of <, >, `@`, and `!` cells on the second and later rows in the `\multirow` which do not appear in the HTML version.

⚠ `\newcolumntype`

- If `\newcolumntype` does not work for HTML, add a simplified column type using `\HTMLnewcolumntype`.

font and alignment

- lwarp detects each of the following, and sets HTML CSS appropriately:

```
>{\centering\arraybackslash}
>{\raggedright\arraybackslash}
>{\raggedleft\arraybackslash}
>{\itshape}
>{\bfseries}
>{\bfseries\itshape}
```

  These may be used with `\newcolumntype`, such as:

```
\newcolumntype{P}[1]{>{\centering\arraybackslash}p{#1}}
```

Rules:

- Doubled `\hlines`, `\midrules`, and vertical rules are supported.

vertical rules

- Vertical rules next to either side of an `@` or `!` column are displayed on both sides of the column.

width and trim

- Width options are honored. Trim options are converted to rounded top corners. Trim corners are not rounded with `@` or `!` columns, and full-width rules ignore trim. When given an optional width, each cell is styled to create the custom border. Without an optional width, the entire row is given a class to assign the standard border.

combined rules

- If you wish to use `\cmidrule` followed by `\bottomrule`, it may be necessary to use:

```
\cmidrule{2-3} \\[-2ex]
\bottomrule
```

  The optional `-2ex` is ignored in HTML, but improves the visual formatting in the print output.

⚠ `\warpprintonly`
⚠ Misplaced `\noalign`

- For `\toprule` and `\bottomrule`, when combined with a `warpprint` or `warpHTML` environment, if a "Misplaced `\noalign`" error occurs, change

```
This & That \endhead
```

  to

```
\warpprintonly{This & That \endhead}
```

  and likewise with the other `\end` headings. Keep the `\endfirsthead` row unchanged, as it is still relevent to HTML output.

Other:

- tabularx ignores the width, but `X` columns do produce paragraph columns or multicolumns.

`longtable` headings

- For longtable, place headings and footings which do not apply to HTML inside `\warpprintonly{}`.

⚠ S columns

- For S columns (from the siunitx package), while producing print output, anything non-numeric must be placed inside { } braces, including commands such as `\multirow`. While producing HTML output, though, anything placed inside braces is not seen by lwarp's tabular handling algorithm. To resolve this problem, make a copy of the row, with one version for print output, containing the extra braces, and another version for HTML output, without the extra braces, such as:

```
\warpprintonly{1 & 2 & {\multirow{2}{2cm}{Text}} & 3 \\}
\warpHTMLonly{1 & 2 & \multirow{2}{2cm}{Text} & 3 \\}
```

⚠ `tabular` inside a `<span>`

- In LaTeX, a `tabular` may be placed inside a `minipage`, but in HTML a `<table>` may not be inside a `<span>`. If this situation is detected, a warning is printed instructing the user to isolate the `<span>` using `\warpprintonly` or the `warpprint` environment.

8.10.2 multirow package

vposn

- Note that recent versions of multirow include a new optional vposn argument.

multirow cells

- For multirow, insert \mrowcell into any empty multi-row cells. This will be a null function for the print output, and is a placeholder for parsing the table for HTML output. An error is generated if this is missed.

```
. . . &  \multirow{2}{.5in}{text}  & . . .
. . . &  \mrowcell                 & . . .
```

colored cells

- The multirow documentation regarding colored cells recommends using a negative number of rows. This will not work with lwarp, so \warpprintonly and \warpHTMLonly must be used to make versions for print and HTML.

with \multicolumn

- See section 429.2 for \multicolumrow.

⚠ \multicolumn & \multirow

lwarp does not support directly combining \multicolumn and \multirow. Use \multicolumnrow instead. To create a 2 column, 3 row cell:

```
\multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}
```

The two arguments for \multicolumn come first, followed by the five arguments for \multirow, many of which are optional, followed by the contents.

⚠ skipped cells

⚠ empty cells

As per \multirow, skipped cells to the right of the \multicolumnrow statement are not included in the source code on the same line. On the following lines, \mcolrowcell must be used for each cell of each column and each row to be skipped. An error is generated if this is missed.

```
. . . &  \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}  & . . .
. . . &  \mcolrowcell          &          \mcolrowcell   & . . .
. . . &  \mcolrowcell          &          \mcolrowcell   & . . .
```

⚠ MathJax

- MathJax does not support multirow, so it is emulated to only print its text on the first row. \multirow works as expected in text tabulars or SVG math.

8.10.3 longtable package

longtable (*Pkg*)

Use one of either \endhead or \endfirsthead for both print and HTML, and use a \warpprintonly macro to disable the other head phrase, and also the \endfoot and \endfirstfoot phrases. (See section 8.10.4 if using threeparttablex.)

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  [ <lastfoot macros> ] \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{
  [ <lastfoot macros> ]   % HTML last footer, without \endfoot
                                          % or \endlastfoot.
}
\end{longtable}
```

⚠ Misplaced \noalign — Use the \warpprintonly macro instead of the warpprint environment. Doing so helps avoid "Misplaced \noalign." when using \begin{warpprint}.

⚠ \kill — \kill is ignored, place a \kill line inside

```
\begin{warpprint} . . .  \end{warpprint}
```

or place it inside \warpprintonly.

⚠ lateximage — longtable is not supported inside a lateximage.

8.10.4 threeparttablex package

threeparttablex (*Pkg*) — threeparttablex is used with longtable and booktabs as follows:

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  \bottomrule \insertTableNotes \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{   % HTML last footer
  \bottomrule
  \UseMinipageWidths     % optional
  \insertTableNotes
  \endlastfoot
}
\end{longtable}
```

table width — The table notes are created using a \multicolumn. By default the width is not specified to the browser, so long table notes can cause the table to be spread out horizontally. For HTML output, lwarp guesses the width of the table depending on the number of columns, then restricts its guess to a min/max range. To use this guess for the width of the table notes, use \UseMinipageWidths before \insertTableNotes. The width is then specified, and in many cases the result is an improvement in overall table layout.

8.10.5 supertabular and xtab packages

supertabular (*Pkg*) — For \tablefirsthead, etc., enclose them as follows:

xtab (*Pkg*)

⚠ Misplaced alignment tab character &

```
\StartDefiningTabulars
\tablefirsthead
. . .
\StopDefiningTabulars
```

See section 8.10.1.

⚠ lateximage — supertabular and xtab are not supported inside a lateximage.

8.10.6 colortbl package

colortbl (*Pkg*)

⚠ row/cell color

Only use \rowcolor and \cellcolor at the start of a row, in that order.

colortbl ignores the overhang arguments.

8.10.7 ctable package

⚠ Misplaced alignment tab character &

Use \StartDefiningTabulars before one or more \ctables, and \StopDefiningTabulars after. These change the meaning of the ampersand & character.

8.10.8 bigdelim package

bigdelim (*Pkg*)

⚠ use \mrowcell

\ldelim and \rdelim use \multirow, so \mrowcell must be used in the proper number of empty cells in the same column below \ldelim or \rdelim, but not in cells which are above or below the delimiter:

```
\begin{tabular}{lll}
<empty> & a & b \\
\ldelim{\{}{3}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
\mrowcell & g & h \\
<empty> & i & j \\
\end{tabular}
```

<-> a b

left { c d
e f
g h

<-> i j

For MathJax, limited emulation is provided which merely prints the delimter and optional text in the first row.

8.11 Floats

8.11.1 Float contents alignment

⚠ figure & table alignment

\centering, etc. are honored in a figure or table if they are the first command inside the float:

```
\begin{table*}
\centering
\caption{A Table}
. . .
```

8.11.2 float, trivfloat, and/or algorithmicx together

float (*Pkg*)
trivfloat (*Pkg*)
algorithmicx (*Pkg*)

⚠ package conflicts

If using \newfloat, trivfloat, and/or algorithmicx together, see section 639.1.

8.11.3 caption and subcaption packages

caption (*Pkg*)
subcaption (*Pkg*)

Package options may cause problems with lwarp, especially if they include curley braces.

If selecting options with braces in `\usepackage` does not work:

```
\usepackage[font={it,small}]{caption}% does not work
```

… try instead selecting the package options before loading lwarp:

```
\PassOptionsToPackage{font={it,small}}{caption}
. . .
\usepackage{lwarp}
. . .
\usepackage{caption}
```

… or try setting package options after the package has been loaded:

```
\usepackage{caption}
\captionsetup{font={it,small}}
```

⚠ numbering

To ensure proper float numbering, set caption positions such as:

```
\captionsetup[figure]{position=bottom}
\captionsetup[subfigure]{position=bottom}
\captionsetup[table]{position=top}
\captionsetup[subtable]{position=top}
```

Similarly for longtable. These positions depend on where the user places the `\caption` command inside each float.

8.11.4 subfig package

subfig (*Pkg*)

⚠ table numbering

To have correct sub table numbers:

```
\usepackage{caption}
\captionsetup[table]{position=top}
```

⚠ lof/lotdepth

At present, the package options for `lofdepth` and `lotdepth` are not working. These counters must be set separately after the package has been loaded.

⚠ horizontal spacing

In the document source, use `\hfill` and `\hspace*` between subfigures to spread them apart horizontally. The use of other forms of whitespace may cause paragraph tags to be generated, resulting in subfigures appearing on the following lines instead of all on a single line.

8.11.5 floatrow package

floatrow (*Pkg*)

⚠ Misplaced alignment tab character &
⚠ subfig package

Use `\StartDefiningTabulars` and `\StopDefiningTabulars` before and after defining macros using `\ttabbox` with a tabular inside. See section 8.10.1.

When combined with the subfig package, while inside a `subfloatrow` `\ffigbox` and `\ttabbox` must have the caption in the first of the two of the mandatory arguments.

⚠ \FBwidth, \FBheight

The emulation of floatrow does not support \FBwidth or \FBheight. These values are pre-set to .3\linewidth and 2in. Possible solutions include:

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.
- Use warpprint and warpHTML environments to select appropriate values for each case.
- Inside a warpHTML environment, manually change \FBwidth or \FBheight before the \ffigbox or \ttabbox. Use \FBwidth or \FBheight normally afterwards; it will be used as expected in print output, and will use your custom-selected value in HTML output. This custom value will be used repeatedly, until it is manually changed to a new value.

8.11.6 keyfloat package

keyfloat (*Pkg*)

⚠ keywrap

If placing a \keyfig[H] inside a keywrap, use an absolute width for \keyfig, instead of lw-proportional widths. (The [H] option forces the use of a minipage, which internally adjusts for a virtual 6-inch wide minipage, which then corrupts the lw option.)

For wrapped figures, overhang and number of lines are ignored.

8.12 KOMA-SCRIPT classes

komascript (*Cls*)

Many features are ignored during the HTML conversion. The goal is source-level compatibility.

\captionformat, \figureformat, and \tableformat are not yet emulated.

⚠ Not fully tested!

Please send bug reports!

Some features have not yet been tested. Please contact the author with any bug reports.

8.13 MEMOIR class

memoir (*Cls*)

⚠ captions

lwarp uses caption, which causes a warning from memoir. This is normal. Adjust captions via caption, instead of memoir.

⚠ options clash

While emulating memoir, lwarp pre-loads a number of packages (section 695.1). This can cause an options clash when the user's document later loads the same packages with options. To fix this problem, specify the options before loading lwarp:

```
\documentclass{memoir}
. . .
\PassOptionsToPackage{options_list}{package_name}
. . .
\usepackage{lwarp}
. . .
\usepackage{package_name}
```

⚠ version numbers

memoir emulates a number of packages, and declares a version date for each which often does not match the date of the corresponding freestanding package. This can cause warnings about incorrect version numbers. Since lwarp is intended to support the freestanding packages, which are often newer than the date declared by memoir, it is hoped that memoir will update and change its emulated version numbers to match.

`\label(bookmark){tag}`

`\label` accepts an optional (`bookmark`) argument, but this is ignored in HTML.

⚠ `comment`

The `comment` environment is from the comment package, and thus requires that the `\begin` and `\end` each be on its own line:

```
\begin{comment}
This is a comment.
\end{comment}
```

`\newcomment`

Comments defined with `\newcomment` use memoir's defintions, and behave as expected, where the `\begin` and `\end` do have to each be on its own line.

⚠ verbatim footnotes

`\verbfootnote` is not supported.

⚠ `\newfootnoteseries`

`\newfootnoteseries`, etc. are not supported.

⚠ page notes

lwarp loads pagenote to perform memoir's pagenote functions, but there are minor differences in `\pagenotesubhead` and related macros.

page notes with cleveref

To add support for pagenotes with cleveref, add:

```
\crefname{pagenote}{page note}{page notes}
\Crefname{pagenote}{Page note}{Page notes}
```

page note `\nameref`

Note that for print mode, `\nameref` print the section name where the page notes are declared in the text, but for HTML it prints the name where the page notes are printed.

⚠ poems

Poem numbering is not supported.

⚠ `verbatim`

The `verbatim` environment does not yet support the memoir enhancements. It is currently recommended to load and use fancyvrb instead.

⚠ glossaries

The memoir glossary system is not yet supported by *lwarpmk*. The glossaries package may be used instead, but does require the glossary entries be changed from the memoir syntax to the glossaries syntax.

⚠ `framewithtitle`, `titledframe`

The custom frame commands in the memoir manual may be emulated by placing the original defintions in the preamble inside `warpprint` environments, and then providing an HTML equivalent:

```
\begin{warpHTML}
\newcommand{\FrameTitle}[2]{%
    \textbf{#2}
}

\newenvironment{framewithtitle}[2][\FrameFirst@Lab\ (cont.)]{%
    \begin{fminipage}{\linewidth}
    \textbf{#2}
    \begin{minipage}{\linewidth}
}
{\end{minipage}\end{fminipage}}
```

```
\newcommand{\TitleFrame}[2]{%
    \par
    \textbf{#1}\par
    \fboxBlock{#2}
}

\newenvironment{titledframe}[2][\FrameFirst@Lab\ (cont.)]{%
    \par
    \textbf{#2}
    \begin{fminipage}{\linewidth}
}
{\end{fminipage}}
\end{warpHTML}
```

8.14 International languages

⚠ section and file names

If using *pdflatex* with the setting `\booltrue{FileSectionNames}`, non-ASCII text in section names can result in corrupted HTML file names. *pdflatex* may be used if setting `\boolfalse{FileSectionNames}`, in which case HTML file numbers will be generated.

For correct HTML file names, use *xelatex*, *lualatex*, or dedicated document classes / engines.

(As of this writing, this warning is only relevent to the kotex package.)

8.15 Miscellaneous packages

8.15.1 verse and memoir

`verse` (*Pkg*)
`memoir` (*Cls*)
`\attrib`

When using verse or memoir, always place a `\\` after each line.

The documentation for the verse and memoir packages suggest defining an `\attrib` command, which may already exist in current documents, but it will only work for print output. lwarp provides `\attribution`, which works for both print and HTML output. To combine the two so that `\attrib` is used for print and `\attribution` is used for HTML:

```
\begin{warpHTML}
\let\attrib\attribution
\end{warpHTML}
```

`\vleftskip` (*Len*)
`\vleftmargini` (*Len*)
`\HTMLvleftskip` (*Len*)
`\HTMLleftmargini` (*Len*)

These lengths are used by verse and memoir to control the left margin, and they may already be set by the user for print output. New lengths `\HTMLvleftskip` and `\HTMLleftmargini` are provided to control the margins in HTML output. These new lengths may be set by the user before any `verse` environment, and persist until they are manually changed again. One reason to change `\HTMLleftmargini` is if there is a wide `\flagverse` in use, such as the word "Chorus", in which case the value of `\HTMLleftmargini` should be set to a wide enough length to contain "Chorus". The default is wide enough for a stanza number.

⚠ spacing

Horizontal spacing relies on *pdftotext*'s ability to discern the layout (`-layout` option) of the text in the HTML-tagged PDF output. For some settings of `\HTMLleftmargini`

⚠ verse margin

or \HTMLleftskip the horizontal alignment may not work out exactly, in which case a label may be shifted by one space. During translation to HTML, the stanza numbers are kept out of the left margin, which would have caused *pdftotext* to shift everything over.

8.15.2 newclude package

newclude (*Pkg*)

⚠ loading

newclude modifies \label in a non-adaptive way, so newclude must be loaded before lwarp is loaded:

```
\documentclass{article}
. . . <font setup>
\usepackage{newclude}
\usepackage[warpHTML]{lwarp}
. . .
```

8.15.3 babel package

babel (*Pkg*)

⚠ \CaptionSeparator

When French is used, the caption separator is changed to a dash. To restore it to a colon, the following may be placed before lwarp is loaded:

```
\renewcommand*{\CaptionSeparator}{:~}
```

punctuation spaces

⚠ customized spacing

Also when French is used, lwarp creates fixed-width space around punctuation by patching \FBcolonspace, \FBthinspace, \FBguillspace, \FBmedkern, \FBthickkern, \FBtextellipsis, and the tilde. If the user's document also changes these parameters, the user's changes should be placed inside a warpprint environment so that the user's changes do not affect the HTML output.

8.15.4 polyglossia package

polyglossia (*Pkg*)

lwarp uses cleveref, which has some limitations when using polyglossia, possibly resulting in the error

```
! Undefined control sequence. . . . \__hook begindocument
```

To test compatibility, add

```
\usepackage{cleveref}
```

near the end of the preamble (as the last package to be loaded), and try to compile the print version. It may be necessary to set

```
\setdefaultlanguage{english}
```

or some other language supported by cleveref, then select other languages using \setotherlanguages.

Once the print version works with cleveref and polyglossia, the HTML version should work as well using lwarp.

8.15.5 todonotes and luatodonotes packages

todonotes (*Pkg*)
luatodonotes (*Pkg*)

The documentation for todonotes and luatodonotes have an example with a todo inside a caption. If this example does not work it will be necessary to move the todo outside of the caption.

8.15.6 fixme

fixme (*Pkg*)

External layouts (`\fxloadlayouts`) are not supported.

⚠ external layouts

Customized layouts are overwritten by lwarp's versions `\AtBeginDocument` in order to provide the HTML conversion. If creating a new layout, see lwarp's changes to provide similar for the new layout, inside a `warpHTML` environment.

User control is provided for setting the HTML styling of the "faces". The defaults are as follows, and may be changed in the preamble after fixme is loaded:

```
\def\FXFaceInlineHTMLStyle{font-weight:bold}
\def\FXFaceEnvHTMLStyle{font-weight:bold}
\def\FXFaceSignatureHTMLStyle{font-style:italic}
\def\FXFaceTargetHTMLStyle{font-style:italic}
```

8.15.7 acro package

⚠ formats

Define acronymn formats using `\textbf` instead of `\bfseries` etc.

8.15.8 chemfig package

If using `\polymerdelim` to add delimiters to a `\chemfig`, wrap both inside a single `lateximage`:

```
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

8.15.9 chemformula package

⚠ chemformula with MathJax

chemformula works best without MathJax. If MathJax is used, `\displaymathother` must be used before `array`, and then `\displaymathnormal` may be used after. (The chemformula package adapts to `array`, but does not know about MathJax, and MathJax does not know about chemformula.)

While using MathJax, `\displaymathother` may also be used for other forms of display and inline math which contain chemformula expressions.

8.15.10 mhchem package

See section 410.

8.15.11 kotex package

kotex (*Pkg*) See section 8.14 regarding *pdflatex* and Korean section names.

⚠ Korean section names

9 Compiling using custom shell commands

lwarp and *lwarpmk* try to make it easy to process print and HTML compilation tasks in most situations. Depending on the operating system, command-line options, TEX engine, and lwarp options, the commands **`lwarpmk print`** and **`lwarpmk html`** are automatically set up to correctly recompile the project. These actions may be overridden using lwarp options, thus allowing the use of packages such as perltex and pythontex.

9.1 Command options

`PrintLatexCmd` (*Opt*)
`HTMLLatexCmd` (*Opt*)

The lwarp options `PrintLatexCmd` and `HTMLLatexCmd` are used to set customized commands to be executed by **`lwarpmk print`** and **`lwarpmk html`**.

`PrintLatexCmd` should be set to shell commands which take `project.tex` and generate `project.pdf`.

`HTMLLatexCmd` should be set to take `project_html.tex` and generate `project_html.pdf`. *lwarpmk* will then take `project_html.pdf` and automatically convert it and generate `project.html`.

9.2 Literal character macros

The lwarp package options are parsed by TEX, and so some characters require the use of a special macro to represent them. See table 10. `\LWRopquote` and `\LWRopseq` may be used to increase operating-system portability. `\jobname` must have `_html` appended for processing HTML. `\space` may be necessary between other macros.

⚠ macro not found

To use these macros, either kvoptions-patch must be loaded before lwarp:

```
\usepackage{kvoptions-patch}
\usepackage[
    PrintLatexCmd={ ... } ,
    HTMLLatexCmd={ ... }
]{lwarp}
```

Table 10: Literal character macros

Character	Macro	Comment
%	\LWRpercent	
$	\LWRdollar	
&	\LWRamp	
%	\LWRhash	
\	\LWRbackslash	
' or "	\LWRopquote	Depends on the operating system.
& or &&	\LWRopseq	Depends on the operating system.
(space)	\space	Forces an extra space.
(jobname)	\jobname	Without file extension.

or \lwarpsetup must be used to set PrintLatexCmd and HTMLLatexCmd:

```
\usepackage[...]{lwarp}
\lwarpsetup{
  PrintLatexCmd=
      {
          latex tm \LWRopseq
          dvips -o tm-pics.ps tm.dvi \LWRopseq
          ps2pdf -dALLOWPSTRANSPARENCY tm-pics.ps \LWRopseq
          pdflatex tm.tex
      } ,
  HTMLLatexCmd=
      {
          latex tm_html \LWRopseq
          dvips -o tm_html-pics.ps tm_html.dvi \LWRopseq
          ps2pdf -dALLOWPSTRANSPARENCY tm_html-pics.ps \LWRopseq
          pdflatex tm_html.tex
      }
}
```

9.3 *latexmk*

latexmk (*Prog*) If *latexmk* is used for a project, it may be easiest to continue using it.

latexmk project.tex would create project.pdf as normal.

latexmk project_html.tex would create project_html.pdf, then

lwarpmk pdftohtml project_html.pdf would take project_html.pdf and convert it to project.html.

sagetex (*Pkg*) *latexmk* may simplify the use of packages such as sagetex.

9.4 perltex package

perltex (*Pkg*) The lwarp package option settings to use perltex would be similar to:

```
\usepackage[
  . . .
  PrintLatexCmd={perltex -latex=pdflatex project.tex} ,
  HTMLLatexCmd={perltex -latex=pdflatex project_html.tex} ,
  . . .
]{lwarp}
```

⚠ "impure" math Place perltex math expressions between `\displaymathother` and `\displaymathnormal`, or `\inlinemathother` and `\inlinemathnormal`. See section 8.7.11.

9.5 pythontex package

pythontex (*Pkg*) An example using pythontex:

```
\usepackage[
  . . .
  PrintLatexCmd={
    pdflatex project.tex \LWRopseq
    pythontex project \LWRopseq
    pdflatex project.tex
  } ,
  HTMLLatexCmd={
    pdflatex project_html.tex \LWRopseq
    pythontex project_html \LWRopseq
    pdflatex project_html.tex
  } ,
  . . .
]{lwarp}
```

Another possibility is to use *latexmk*, placing the **`latexmk . . .`** commands in the `PrintLatexCmd` and `HTMLLatexCmd` options. While using these options, the lwarp option `latexmk` would not be used.

⚠ "impure" math
⚠ HTML look-alike

No attempt has yet been made to make pythontex robust with HTML output. Some math objects must be surrounded by `\displaymathother` ... `\displaymathnormal`, or `\inlinemathother` ... `\inlinemathnormal`. Displays of code may have to be enclosed inside a `lateximage` environment to prevent <, > and similar from being interpreted by the browser as HTML entities.

9.6 sympytex package

sympytex (*Pkg*) For sympytex:

```
\usepackage[
  . . .
  PrintLatexCmd={
    pdflatex project.tex \LWRopseq
    python project.sympy \LWRopseq
    pdflatex project.tex
  } ,
  HTMLLatexCmd={
    pdflatex project_html.tex \LWRopseq
    python project_html.sympy \LWRopseq
    pdflatex project_html.tex
  } ,
  . . .
]{lwarp}
```

Also see the warnings for pythontex, above.

9.7 Other packages

rterface (*Pkg*) Other packages such as rterface would be set up similar to pythontex, and the same warnings would apply.

9.8 *make* program

make (*Prog*) To use lwarp with the *make* program, have the `makefile` take `project.tex` and generate the print version `project.pdf`, as normal. `\usepackage{lwarp}` must be used, and it generates `lwarpmk.conf` when the print version is created.

To generate HTML, first have `project_html.tex` be compiled to generate `project_html.pdf`. This must be in PDF format. Finally, have `project_html.pdf` be converted to HTML using **`lwarpmk pdftohtml project_html.pdf`**, and convert SVG math with **`lwarpmk limages`**.

9.9 UTF-8 locale

⚠ UTF-8 locale *lwarpmk* uses the *texlua* program, which sets the "locale" to "C", including for external operating-system calls such as when executing **`lwarpmk html`**. In some cases, an external program called from the user's document may require the use of a UTF-8 "locale". For UNIX-related operating systems, it may be required to use lwarp's custom compilation options to add a locale change:

```
\usepackage{lwarp}[
  PrintLatexCmd={
    env LC_CTYPE=en_US.UTF-8
      xelatex -shell-escape project.tex
  }
  HTMLLatexCmd={
    env LC_CTYPE=en_US.UTF-8
      xelatex -shell-escape project_html.tex
  }
]
```

`ditaa` (*Pkg*) The only example seen so far where this is required is the ditaa package, where the locale change allows the use of UTF-8 with XeLaTeX and ditaa. To use LuaLaTeX instead, the locale change would have to be made inside the ditaa package where its calls the *ditaa* program.

10 EPUB conversion

lwarp does not produce EPUB documents, but it may be told to modify its HTML output to greatly assist in the conversion. An external program may then be used to finish the conversion to EPUB.

<meta> author

To assign the author's name for regular lwarp HTML files, and also for the EPUB, use `\HTMLAuthor {⟨name⟩}`. This assigns the name to the `<meta>` author element. It may be set empty, and it defaults to `\theauthor`.

A special boolean is provided to simplify the process of converting lwarp HTML output to EPUB:

`FormatEPUB`

`FormatEPUB` (*bool*)
Default: `false`

`FormatEPUB` changes HTML output for easy EPUB conversion via an external program. Removes per-file headers, footers, and nav. Adds footnotes per chapter/section.

To help convert lwarp HTML output to EPUB, add

```
\booltrue{FormatEPUB}
```

to the project's source preamble after `\usepackage{lwarp}`. The EPUB version of the document cannot co-exist with the regular HTML version, so

Enter ⇒ **`lwarpmk cleanall`**

Enter ⇒ **`lwarpmk html`**

Enter ⇒ **`lwarpmk limages`**

to recompile with the `FormatEPUB` boolean turned on. Several changes are then made to the HTML output:

- Headers, footers, and navigation are removed at file splits.
- Any accumulated footnotes are printed at the bottom of each section.

Calibre

The resulting files will be ready to be loaded into an EPUB conversion program, such as the open-source program *Calibre* (`https://calibre-ebook.com/`).

⚠ search order

The EPUB conversion program must know what order the files are included. For lwarp projects, set the EPUB conversion software to do a breadth-first search of the files. For *Calibre*, this option is found in

Preferences → Plugins → File type plugins → HTML to Zip

⚠ encoding

Check the box Add linked files in breadth first order. Set the document encoding as `utf-8`, which is what lwarp generates for HTML, even if the original printed document uses some other encoding.

⚠ section breaks

The EPUB-conversion program must also know where the section breaks are located. For a list of lwarp's section headings, see table 12. For example, an `article` class document would break at `\section`, which is mapped to HTML heading level

<h4>, whereas a book class document would break at \chapter, which is HTML heading level <h3>. For *Calibre*, this option is found in

Preferences → Conversion (Common Options) → Structure Detection → Detect chapters at (XPath expression)

Select the "magic wand" to the right of this entry box, and set the first entry

Match HTML tags with tag name:

to "h4". (Or "h3" for document classes with \chapters.) The Detect chapters at field should then show

//h:h4 — or — **//h:h3**

This option is also available on the main tool bar at the Convert books button.

Once these settings have been made, the lwarp-generated HTML files may be loaded by *Calibre*, and then converted to an EPUB.

MathJax support

MathJax may be used in EPUB documents. Some e-readers include MathJax, but any given reader may or may not have a recent version, and may or may not include extensions such as support for siunitx.

lwarp adds some modifications to MathML to support equations numbered by chapter. These modifications may not be compatible with the e-reader's version of MathJax, so lwarp requests that a known version be loaded instead. In some cases chapter numbering of equations still doesn't work.

Until math support in EPUB documents is improved, it is recommended to use SVG images instead of MathJax, especially for equations numbered by chapter, or where siunitx support is important.

11 Word-processor conversion

lwarp may be told to modify its HTML output to make it easier to import the HTML document into a word processor. At the time of this writing, it seems that LibreOffice works best at preserving table layout, but it still has some limitations, such as an inability to automatically assign figure and table frames and captions according to user-selected HTML classes. lwarp provides some assistance in locating these frame boundaries, as shown below.

11.1 Activating word-processor conversion

A special boolean is provided to simplify the process of converting lwarp HTML output to EPUB:

`FormatWP`

`FormatWP` (*bool*)
Default: `false`

Changes HTML output for easier conversion by a word processor. Removes headers and nav, prints footnotes per section, and also forces single-file output and turns off HTML debug comments. Additionally, honors the booleans `WPMarkFloats`, `WPMarkMinipages`, `WPMarkTOC`, and `WPMarkLOFT`.

To help modify lwarp HTML output for easier import to a word processor, add

```
\booltrue{FormatWP}
```

to the project's source preamble after lwarp is loaded. The following changes are formatting adjustments then made to the HTML output:

- If using a class without chapters, `\section` and lower are shifted up in level for the HTML heading tags. The CSS has not been changed, so the section heading formats will not match the normal HTML output, but when imported to *LibreOffice Writer* the higher section headings will import as **Heading 1** for the title, **Heading 2** for `\section`, etc.
- Headers, footers, and navigation are removed at file splits.
- Any accumulated footnotes are printed at the bottom of each section.
- Forces single-file output.
- Turns off HTML debugging comments. These are comments appearing inside the HTML code, marking the opening/closing of sections and `<div>`s, but they are no longer useful when the document has been imported into a word processor.
- An additional `<div>` with an `id` encapsulates each float and minipage, which on import into *LibreOffice Writer* causes a thin frame to appear around the text block for each.
- Float captions are given an explicit italic formatting.
- Tabular rule borders are made explicit for *LibreOffice Writer*. LibreOffice displays a light border around each cell while editing, even those which have

no border when printed, and lwarp also uses a light border for thin rules, so it will be best to judge the results using the print preview instead of while editing in LibreOffice.

- `\includegraphics` and SVG math width and height are made explicit for LibreOffice.
- `\hspace` is approximated by a number of `\quad`s, and rules are approximated by a number of underscores.
- Explicit HTML styles are given to:
  - `\textsc`, etc.
  - `\underline`, soul and ulem markup.
  - `center`, `flushleft`, `flushright`.
  - `\marginpar`, keyfloat, sidenotes, floatflt, and wrapfig.
  - fancybox `\shadowbox`, etc.
  - The LaTeX and TeX logos.
- Honors several booleans:

  WPMarkFloats: Marks the begin and end of floats.

  WPMarkMinipages: Marks the begin and end of minipages.

  WPMarkTOC: Marks the location of the Table of Contents.

  WPMarkLOFT: Marks the locations of the List of Figures/Tables.

  WPMarkMath: Prints LaTeX math instead of using images.

  WPTitleHeading: Adjusts title and section headings.

  Several of these may be used to add markers to the HTML text which help determine where to adjust the word processor document after import.

11.2 Additional modifications

`WPMarkFloats`

`WPMarkFloats` (*bool*)
Default: `false`

Adds

```
=== begin table ===
. . .
=== end ===
```

or

```
=== begin figure ===
. . .
=== end ===
```

around floats while formatting for word processors. This helps identify boundaries of floats to be manually converted to word-processor frames and captions.

`WPMarkMinipages`

`WPMarkMinipages` (*bool*)
Default: `false`

Adds

```
=== begin minipage ===
. . .
=== end minipage ===
```

around minipages while formatting for word processors. This helps identify boundaries of minipages to be manually converted to word-processor frames.

`WPMarkTOC`

`WPMarkTOC` (*bool*)
Default: `true`

While formatting for word processors, adds

```
=== table of contents ===
```

where the Table of Contents would have been. This helps identify where to insert the actual TOC.

If set `false`*, the actual toc is printed instead.*

`WPMarkLOFT`

`WPMarkLOFT` (*bool*)
Default: `false`

While formatting for word processors, adds

```
=== list of figures ===    and/or
=== list of tables ===
```

where each of these lists would have been. This helps identify where to insert the actual lists.

If set `false`*, the actual lists are printed instead.*

`WPMarkMath`

---

siunitx
`WPMarkMath` (*bool*)
Default: `false`
`TeXMaths` (*Prog*)

While formatting for word processors, prints math as LaTeX code instead of creating SVG images or MathJax. This is useful for cut/paste into the *LibreOffice Writer TeXMaths* extension.

When using the siunitx package, enter

```
\usepackage{siunitx}
```

in the *TeXMaths* preamble. Equation numbering is problematic for $\mathcal{AMS}$ math environments.

`WPTitleHeading`

`WPTitleHeading` (*bool*)
Default: `false`
section headings

While formatting for word processors, `true` sets the document title to `<h1>`, which is expected for HTML documents, but also causes the lower-level section headings to start at **Heading 2** when imported into LibreOffice. Set to `false` to cause the title to be plain text, and the section headings to begin at **Heading 1**.

Table 11: Section HTML headings for word-processor conversion

	HTML headings*			
	With \chapter		Without \chapter	
	WPTitleHeading		WPTitleHeading	
Section	true	false	true	false
Title	<h1>	plain	<h1>	plain
\book				
\part	<h2>	<h1>	<h2>	<h1>
\chapter	<h3>	<h2>	—	—
\section	<h4>	<h3>	<h3>	<h2>
\subsection	<h5>	<h4>	<h4>	<h3>
\paragraph	<h6>	<h5>	<h5>	<h4>
\subparagraph		<h6>	<h6>	<h5>

* For default depths when not FormatWP, see table 12 on page 201.

See table 11 on page 185.

11.3 Recommendations

TOC, LOF, LOT

For use with *LibreOffice Writer*, it is recommended to:

1. Set \booltrue{FormatWP}
2. Set \booltrue{WPMarkTOC} and \boolfalse{WPMarkLOFT}
3. Use lwarp to generate the HTML document.
4. Copy/paste from the HTML document into an empty *LibreOffice Writer* document.
5. Manually insert a LibreOffice TOC in the LibreOffice document.
6. Manually add frames around each float, adding a caption which is cut/pasted from each float's simulated caption.
7. Manually create cross references.

This process yields a document with an actual LibreOffice Table of Contents, but a simulated List of Figures and List of Tables.

siunitx

For siunitx, remember to adjust the preamble as mentioned above.

LO view border options

LibreOffice has options in the **View** menu to turn on/off the display of thin borders around table cells and text objects.

11.4 Limitations

Floats and captions are not explicitly converted to LibreOffice floats with their own captions. Floats are surrounded by a thin frame in the LibreOffice editor, and may be marked with `WPMarkFloats`, but are not given a proper LibreOffice object frame. Captions are given an explicit italic formatting, but not a proper LibreOffice paragraph style.

Cross references are not actual LibreOffice linked cross references.

The List of Figures and List of Tables are not linked. The pasted pseudo LOF and LOT match the numbering of the LaTeX and HTML versions.

Equation numbering is not automatic, but the equation numbers in SVG math will match the LaTeX and HTML output. SVG math is recommended when using the $\mathcal{AMS}$ environments, which may have multiple numbered equations per object.

As of when last checked, LibreOffice ignores the following:

- Minipage alignment.
- Tabular cell vertical alignment.
- Image rotation and scaling.
- Rounded border corners, which are also used by:
  - `\textcircled`
  - booktabs trim
- `\hspace` and rules, also used by algorithmic.
- Coloring of text decorations, used by soul and ulem.
- Overline text decoration, used by romanbar.

LibreOffice also has limitations with frames and backgrounds:

- Multiple lines in an object are framed individually instead of as a whole.
- Nested frames are not handled correctly.
- Images inside boxes are not framed correctly.
- Spans with background colors and frames are not displayed correctly.

12 Modifying lwarp

locating something

To quickly find the source for a package in `lwarp.dtx`, search for `*packagename`, such as `*siunitx`.

Likewise, to quickly find the source for a file in `lwarp.dtx`, search for `*filename`, such as `*lwarp.css`.

Purely text-based packages probably will work as-is when generating HTML.

Look to existing code for ideas on how to expand into new code.

image of TEX output

An environment may be converted to a `lateximage` then displayed with an image of the resulting LATEX output. See section 93 for an example of the `picture` environment.

css classes

To create a custom HTML block or inline CSS class, see section 52.10.

print/HTML macros

To create print and HTML versions of the same macro or environment, see section 36.

⚠ TEX boxes

Any TEX boxes must be undone, as SVG math or `lateximages` require `\newpage`, which will not work in a TEX box.

12.1 Creating a development system

The following creates a local development system for lwarp on a TeXLive system in a UNIX-like environment. Doing so allows anything requesting lwarp to use the development version instead of whichever version is installed in TeXLive.

Create a development directory:

Place into this directory `lwarp.dtx` and `lwarp.ins`.

To create `lwarp.sty`, execute

Enter ⇒ **`pdflatex lwarp.ins`**

which creates `lwarp.sty` and several hundred additional `lwarp-*.sty` files for the various packages which are supported.

To create the initial documentation `lwarp.pdf`, execute

Enter ⇒ **`pdflatex lwarp.dtx`**

To make the development files visible to other projects:

Create the directory

```
/usr/local/texlive/texmf-local/tex/latex/local/lwarp
```

Inside this directory, create the file `update`, containing:

```
rm lwarp-*.sty
ln -s /path_to_dev_directory/lwarp*.sty .
ln -s /path_to_dev_directory/lwarp_baseline_marker.png .
ln -s /path_to_dev_directory/lwarp_baseline_marker.eps .
mktexlsr
```

Run ./update now, and whenever a new lwarp-* package is added.

To make the development version of *lwarpmk* visible to other projects:

```
cd /opt
ln -s /usr/local/texlive/texmf-local/bin/x86_64-linux texbin_local
cd texbin_local
ln -s ../../scripts/lwarp/lwarpmk.lua lwarpmk
cd /usr/local/texlive/texmf-local/scripts/
mkdir lwarp
cd lwarp
ln -s /path_to_dev_directory/lwarpmk.lua lwarpmk
```

Verify that the correct version is found with

Enter ⇒ **`which lwarpmk`**

To make the local versions visible to the shell:

Paths must be set by the shell startup, such as in `.bashrc` and `.cshrc`:

In `.bashrc`:

```
PATH=/opt/texbin_local:/opt/texbin:$PATH
```

In `.cshrc`:

```
setenv PATH ${HOME}/bin:/opt/texbin_local:/opt/texbin:${PATH}
```

To fully compile the lwarp documentation and indexes:

```
pdflatex lwarp.ins
pdflatex lwarp.dtx
pdflatex lwarp.dtx                          <if necessary>
makeindex -s gglo.ist -o lwarp.gls lwarp.glo   <indexes>
splitindex lwarp.idx - -s gind.ist
pdflatex lwarp.dtx
pdflatex lwarp.dtx                          <if necessary>
makeindex -s gglo.ist -o lwarp.gls lwarp.glo   <indexes>
splitindex lwarp.idx - -s gind.ist              <again>
pdflatex lwarp.dtx
pdflatex lwarp.dtx                          <if necessary>
```

(The second round of index processing is required to fully resolve the final Index of Indexes.)

To make it easier to update the documentation after a minor change, it is useful to create a command script called `make_index`, containing:

```
makeindex -s gglo.ist -o lwarp.gls lwarp.glo
splitindex lwarp.idx -- -s gind.ist
```

⚠ references

Note that Index of Indexes and the cross-references to the indexes may not be correct until the above has been accomplished.

12.2 Modifying a package for lwarp

If a class loads additional packages, it will be required to modify the class for lwarp, since lwarp must be loaded before most other packages.

To work with lwarp, a class must first set up anything which replicates the functions of the basic LaTeX classes, load any required fonts, then load lwarp, then finally load and adjust any other required packages.

When creating HTML, lwarp redefines the `\usepackage` and `\RequirePackage` macros such that it first looks to see if a `lwarp-<packagename>.sty` version exists. If so, the lwarp version is used instead. This modular system allows users to create their own versions of packages for lwarp to use for HTML, simply by creating a new package with a `lwarp-` prefix. If placed in the local directory along with the source code, it will be seen by that project alone. If placed alongside the other `lwarp-` packages where TeX can see it, then the user's new package will be seen by any documents using lwarp. (Remember `mktexlsr` or `texhash`.)

An `lwarp-<packagename>.sty` package is only used during HTML generation. Its purpose is to pretend to be the original package, while modify anything necessary to create a successful HTML conversion. For many packages it is sufficient to simply provide nullified macros, lengths, counters, etc. for anything which the original package does, while passing the raw text on to be typeset. See the pre-existing `lwarp-` packages for examples.

Anything the user might expect of the original package must be replaced or emulated by the new `lwarp-` package, including package options, user-adjustable counters, lengths, and booleans, and conditional behaviors. In many of these packages, most of the new definitions have a "local" prefix according to the package name, and `@` characters inside the name, which hides these names from the user. In most cases these macros will not need to be emulated for HTML output. Only the "user-facing" macros need to be nullified or emulated.

Each `lwarp-*` package should first call either of:

```
\LWR@ProvidesPackageDrop
  — or —
\LWR@ProvidesPackagePass
```

If "`Drop`"ped, the original print-version package is ignored, and only the `lwarp-` version is used. Use this where the original print version is useless for HTML. If "`Pass`"ed, the original package is loaded first, with the user-supplied options, then the `lwarp-` version continues loading as well. See section 457 (ntheorem) for an example of selectively disabling user options for a package. Use this when HTML output only requires some modifications of the original package. For a case where the original package is usable without changes, there is no need to create a `lwarp-` version.

12.2.1 Adding a package to the lwarp.dtx file

When adding a package to `lwarp.dtx` for permanent inclusion in lwarp, provide the `lwarp-<packagename>` code in `lwarp.dtx`, add its entry into `lwarp.ins`, and also remember to add

```
\LWR@loadafter{<packagename>}
```

to `lwarp.dtx` in section 20.1. This causes lwarp to stop with an error if packagename is loaded before lwarp. Finally, add an entry in table 2, Supported packages and features, and also the Updates section.

12.3 Modifying a class for lwarp

If a class loads additional packages, it will be required to modify the class for lwarp, since lwarp must be loaded before most other packages.

To work with lwarp, a class must first set up anything which replicates the functions of the basic LaTeX classes, load any required fonts, then load lwarp, then finally load and adjust any other required packages.

12.4 Testing lwarp

Compiling `lwarp.ins` generates all the `*.sty` files for lwarp. It can be useful to create additional `*.ins` files to be able to recompile only the pieces which have changed.

compiling individual packages

`core.ins` (*file*)

For example, to be able to recompile the lwarp core alone, copy `lwarp.ins` to `core.ins`, then modify `core.ins` to only compile:

```
\generate{
\file{lwarp.sty}{\from{lwarp.dtx}{package}}
}
```

For individual packages, create `packagename.ins`, set to compile only:

```
\generate{
\file{lwarp-packagename.sty}{\from{lwarp.dtx}{packagename}}
}
```

When changes have been made, test the print output before testing the HTML. The print output compiles faster, and any errors in the printed version will be easier to figure out than the HTML version.

compiling css and other generated files

Remember that the configuration files are only rewritten when compiling the printed version of the document.

When changing the source to *lwarpmk* or a CSS file in `lwarp.dtx`:

1. Change the source in `lwarp.dtx`.
2. `pdflatex lwarp.ins` -or- `pdflatex core.ins`
3. `pdflatex lwarp.dtx`
4. If modifying *lwarpmk* the new version should now be active.
5. If modifying CSS files or other files generated by lwarp:
   (a) For the document, `lwarpmk print` to update the CSS files in the project.
   (b) Reload the HTML document to see the effect of the new CSS files.

Sometimes it is worth checking the <project>_html.pdf file, which is the PDF containing HTML tags. Also, <project>_html.html has the text conversion of these tags, before the file is split into individual HTML files.

It is also worth checking the browser's tools for verifying the correctness of HTML and CSS code.

12.5 Modifying *lwarpmk*

lwarpmk (*Prog*)
lwarpmk.lua (*file*)

In most installations, lwarpmk.lua is an executable file located somewhere the operating system knows about, and it is called by typing **lwarpmk** into a terminal.

A project-local copy of lwarpmk.lua may be generated, modified, and then used to compile documents:

1. Add the lwarpmk option to the lwarp package.
2. Recompile the printed version of the document. The lwarpmk option causes lwarp to create a local copy of lwarpmk.lua
3. The lwarpmk option may now be removed from the lwarp package.
4. Copy and rename lwarpmk.lua to a new file such as mymake.lua.
5. Modify mymake.lua as desired.
6. If necessary, make mymake.lua executable.
7. Use mymake.lua instead of lwarpmk.lua.

13 Troubleshooting

13.1 lwarp package error conditions and warnings

lwarp tests for a number of error conditions and prints appropriate warnings. The following is a summary of these conditions.

13.1.1 Configuration file `lwarpmk.conf`

File does not exist: The configuration file must exist for lwarpmk.

Incorrect `Unix` /`Windows` selection: The operating system which was detected by lwarp. So far only `Unix` and `Windows` are supported.

Incorrect delimiter characters. Older versions of *lwarpmk* used a different delimiter.

Source name is set to `lwarp`: lwarp has recently been recompiled in this directory, which overwrote the project's configuration files. This also occurs if *lwarpmk* is executed in lwarp's source directory.

Incorrect operating system: The configuratio file was set for a different operating system, perhaps due to sharing in a collaborative project.

Outdated configuration files: lwarp has been updated since this projects was last compiled. If there appears to be a valid print command in the file, lwarpmk displays this to instruct the user how to recompile the print version, which then updates the configuration files.

The designated source file does not exist: For whatever reason…

Unknown engine: lwarp cannot determing which engine is being used. Supported are DVI LaTeX, PDF LaTeX, XeLaTeX, LuaLaTeX, and upLaTeX.

13.1.2 Image generation with `lwarpmk limages`

"Wait a moment for the images to complete before reloading page.":

Images are generated by background tasks. If the document is reloaded before these tasks are complete, some images may not yet be generated. *lwarpmk* tries to wait for background tasks to complete before exiting.

HTML version does not exist: Images are extracted from the HTML version, which must be compiled before images are generated.

`*-images.txt` does not exist: This file tells which images to extract from the HTML file. If the file does not exist, it may be that no SVG math or `lateximages` were used. If so, **`lwarpmk limages`** is not necessary.

Cross references are not correct: The document must have up-to-date cross references to locate the images to extract. A number of conditions may cause incorrect cross references.

"WARNING: Images will be incorrect.": An image reference was not found. Recompile.

`lwarpmk epstopdf *` or `lwarpmk pdftosvg *`: Errors if filenames are not found.

13.1.3 Default bitmapped font

lwarp requires the use of a vector font. If lwarp detects that the document uses the default COMPUTER MODERN font, and the cm-super package is not installed, it is assumed that the font is bitmapped. An error is generated, along with the recommendation to install cm-super or use lmodern.

13.1.4 Packages

Loaded before lwarp: Some packages and classes must be loaded before lwarp. These include input and font encoding, morewrites and newclude, and a number of CJK-related packages and classes.

Loaded after lwarp: Most packages which are modified by lwarp must be loaded after lwarp.

Loaded never: Some packages do not work with lwarp. An error is generated, along with a list of alternatives to consider.

Specific packages: Some packages enforce a specific load order vs. certain other packages.

Patching error: lwarp tries to patch some packages using xpatch. If the original package has been updated more recently than lwarp, a patch may not work. It may be necessary to use an older version of the package until lwarp is updated.

longtable: lwarp's longtable package issues detailed error messages regarding the use of the table headers and footers.

polyglossia: If used, an informative message is printed to instruct the user to be sure to set a language, without which an error will occur.

babel or polyglossia: An informative message is printed to note that not all langauges are supported by cleveref.

13.1.5 Compiling

`SideTOCDepth` < `FileDepth`: A warning is displayed if these counters are set such that the sideTOC will not be able to access all pages of the website.

Filenames: lwarp may generate file names from section names. While doing so, the filenames are simplified, and special characters and math are removed. If this process generates a duplicate filename, and error is generated, describing the filename and which section name generated it. A warning is issued if dollar-delimited math is used. Parenthesis-delimited math is recommended instead.

⚠ HTML corrupted **Multirow:** When `\multirow` or `\multicolrow` are used, `\mrowcell` or `\mcolrowcell` must be placed in the appropriate cells to avoid corrupted HTML output.

(width,height) missing a comma: `\makebox` and `\framebox` can accept a parenthesis-delimited width and height, which must be separated by a comma.

"Load graphicx or graphics for improved SVG math baselines.": SVG math sizing and baselines are improved if either of these packages are used.

"Load graphicx or graphics for improved XeTeX logo.": If these packages are loaded, the XeLaTeX logo can use the reversed "E".

"It is recommended to use `[width=xx\linewidth]` instead of `[scale=yy]` ": Browser support of `scale` does not have the same effect as in LaTeX.

13.2 Using the lwarp package

The following address problems which may occur, and possible solutions to each.

Also see:
Section 7.11: Commands to be placed into the `warpprint` environment
Section 8: Special cases and limitations

⚠ HTML corrupted **Text is not converting correctly / corrupted HTML tags:**

- Font-related UTF-8 information must be embedded in the PDF file. See section 7.4 regarding bitmapped vs. vector fonts.
- See section 8.2.1 regarding HTML entities and the characters &, <, and >.

⚠ dotlessj **Dotless j (`\j`):** See section 7.4 regarding cmap, mmap.

Undefined HTML settings:

- See the warning regarding the placement of the HTML settings at section 7.6.

Tabular problems: See section 8.10.1.

Obscure error messages:

Print first: Be sure that a print version of the document compiles and that your document's LaTeX code is correct, before attempting to generate an HTML version.

`\end{warpHTML}, \end{warpprint}, \end{warpall}, \end{warpMathJax}`:

⚠ `warpHTML`, `warpprint`, `warpMathJax`, `warpall`

Each of these must be without any other characters on the same line.

"`Runaway argument? File ended while scanning use of \next`: Don't use `warpHTML`, `warpprint`, `warpall`, or `warpMathJax` inside itself.

Options clash: If using memoir, see section 8.13.

"Missing `\begin{document}`.": Some packages require that their options be specified before lwarp is loaded, or via the package's setup macro, especially if these options include the use of braces. See section 8.1.

"No room for a new `\write`.": Before `\usepackage{lwarp}`, add:

```
\usepackage{morewrites}
\morewritessetup{allocate=10}
```

"! TeX capacity exceeded, sorry [text input levels=15].": Packages were nested too many levels deep. Locate the file `texmf.cnf` for your distribution, and add the line

```
max_in_open = 30
```

"Missing $ inserted.": If using a filename or URL in a footnote or `\item`, escape underscores with `\_`.

"Label(s) may have changed. Rerun to get cross-references right.":
This warning may repeat endlessly if a math expression is used in a caption. Simple math expressions such as $X=1$ may be replaced with

```
\textit{X}\,=\,1
```

"Temporary page! LaTeX was unable to guess the total number of pages ...": Harmless. Recompile the document one more time.

"Leaders not followed by proper glue": This can be caused by a missing `l@<floattype>` or `l@<sectiontype>` definition. See lwarp's definitions for examples.

"Improper \prevdepth": `lateximages` and SVG math require `\newpage`, which cannot work inside TEX boxes or `\ensuremath`. Anything using `\newsavebox`, `\newbox`, `lrbox`, `\savebox`, `\hbox`, `\vbox`, `\usebox`, `\sbox`, etc., must be modified to work without box commands.

If you find something using `\ensuremath`, have it temporarily set:

```
\LetLtxMacro\@ensuredmath\LWR@origensuredmath
```

inside a group first.

`LWR@texboxdepth`

As a stop-gap measure, you may wish to try incrementing the counter `LWR@texboxdepth` before the problematic macro, and then decrementing it after. Doing so tells lwarp to avoid using a `\newpage` inside the macro, which may avoid this error.

⚠ macros in section, table, figure names

Also, custom macros which appear inside a section, figure, or table name should be made robust since they appear inside the `.toc`, `.lof`, or `.lot` files. Use `\newrobustcmd` or `\robustify` from etoolbox, xparse, etc.

⚠ BibTeX

If using BibTeX, see section 8.6.9.

⚠ polyglossia

"! Undefined control sequence. . . . __hook begindocument": See section 8.15.4 if using polyglossia.

⚠ custom macros for environments

"\begin{equation} ended by \end{document}": Do not use custom macros such as `\beq` and `\eeq` to replace

```
\begin{equation}
. . .
\end{equation}
```

⚠ `\LWR@formatted`

"Misplaced \omit": If using `\LWR@formatted` to define new macros for print and HTML modes, see section 36 regarding `\LWR@expandableformatted`.

"Token not allowed in a PDF string": This hyperref warning appears while creating the print-mode document, not HTML. A low-level macro is being used in a section name which appears in the PDF bookmarks. hyperref removes this macro from the bookmark, and warns of doing so. To avoid this warning, use `\pdfstringdefDisableCommands` in the preamble to define simplified replacement macros for each, or use `\texorpdfstring` in the `\section` or related macro to declare what to use for the TEX text, v.s. the PDF bookmark. See the hyperref manual.

⚠ quote character

"Command \textquoteright invalid in math mode": This can occur when the document source has math containing the slanted quote ' character, instead of using the upright quote ' character.

⚠ "impure" math objects

Complicated objects inside math: Some objects, such as Ti*k*Z, may not compile in lwarp's normal math emulation. Insert

`\displaymathother` —*or*— `\inlinemathother`

before the math, and then

`\displaymathnormal` —*or*— `\inlinemathnormal`

when displaying "normal" math. See section 8.7.11.

Slow compliation of math objects: Complicated math objects can also cause problems with `alt` tags, resulting in very slow compilation, large `alt` tags, and possible crashes. Use `\inlinemathother` ... `\inlinemathnormal` or `\displaymathother` ... `\displaymathnormal` around the math expression.

⚠ MathJax **Incorrect MathJax:** Some objects do not convert to MathJax. Use `\displaymathother` before these objects, then `\displaymathnormal` to return to "normal" display math. See section 8.7.11.

Missing sections: See section 7.6 regarding the `FileDepth` and `SideTOCDepth` counters, and the use of `\tableofcontents` in the home page.

Misnumbered footnotes from section headings: See section 8.5.4.

Missing HTML files:

- See the warning regarding changes to the HTML settings at section 7.6.
- Ensure that the filenames are unique after math and short words are removed. See `FileSectionNames` at section 7.6.

Missing / incorrect cross-references:

- Use `lwarpmk again` followed by `lwarpmk html` or `lwarpmk print` to compile the document one more time.
- labels
  ⚠ label characters
  Labels with special characters may be a problem. It is best to stick with alpha-numeric, hyphen, underscore, and perhaps the colon (if not French).

  `\nameref` refers to the most recently-used section where the `\label` was defined. If no section has been defined before the `\label`, the link will be empty. Index entries also use `\nameref` and have the same limitation.

  `\nameref`
  ⚠ empty link
- ⚠ cleveref page numbers
  cleveref and varioref are supported, but printed page numbers do not map to HTML, so a section name or a text phrase are used for `\cpageref` and `\cpagerefrange`. This phrase includes `\cpagerefFor`, which defaults to "for".

  Ex:

  ```
  \cpageref{tab:first,tab:second}
  ```

  in `html` becomes:

  "pages **for** table 4.1 and **for** table 4.2"

  See `\cpagerefFor` at page 734 to redefine the message which is printed for page number references.

BibTeX errors with `\etalchar`: See section 8.6.9.

Malformed URLs: Do not use the `%` character between arguments of `\hyperref`, etc., as this character is among those which is neutralized for inclusion in HTML URLs.

Em-dashes or En-dashes in listing captions and titles:

Use XeLaTeX or LuaLaTeX.

Floats out of sequence:

Mixed "Here" and floating: Floats [H]ere and regular floats may become out of order. `\clearpage` if necessary.

Caption setup: With `\captionsetup` set the positions for the captions `above` or `below` to match their use in the source code.

Images are appearing in strange places:

- When images are added or removed, Enter **`lwarpmk limages`** to refresh the `lateximage` images.

SVG images:

⚠ adding/removing

When a math expression, `picture`, or Ti*k*Z environment is added or removed, the SVG images must be re-created by entering **`lwarpmk limages`** to maintain the proper image-file associations. Inline SVG math may be hashed and thus not need to be recreated, but display math and objects such as Ti*k*Z may move to new image numbers when the document is changed.

recompile first

Before attempting to create the SVG image files, *lwarpmk* verifies that the HTML version of the document exists and has correct internal image references.[16] If it is necessary to recompile the document's HTML version one more time, *lwarpmk* usually will inform the user with an error message, but there are some conditions which cannot be detected, so the user should watch for the LaTeX recompile warnings.

⚠ HTML instead of images

If HTML appears where an SVG image should be, recompile the document one more time to get the page numbers back in sync, then remake the images one more time.

⚠ page counter

Incorrect SVG images will also occur if the document changes the `page` counter:

```
\setcounter{page}{<value>}
```

The `page` counter must *not* be adjusted by the user.

⚠ Lots of files!

Expressing math as SVG images has the advantage of representing the math exactly as LaTeX would, but has the disadvantage of requiring an individual file for each math expression. For inline math, and some other objects, lwarp uses an MD5 hash on its LaTeX source to combine multiple instances of identical inline expressions into a single image file, but display math and other environments such as `picture` and Ti*k*Z require one image file each. For a document with a large amount of math, see section 5.5 to use MathJax instead.

Plain-looking document:

- The document's CSS stylesheet may not be available, or may be linked incorrectly. Verify any `\CSSFilename` statements point to a valid CSS file.

⚠ HTML corrupted

Broken fragments of HTML:

- Check the PDF file used to create HTML to see if the tags overflowed the margin. (This is why such large page size and margins are used.)

Changes do not seem to be taking effect:

[16] This becomes important when dealing with a document containing thousands of images.

- Be sure to `lwarpmk clean`, recompile, then start by reloading the home page. You may have been looking at an older version of the document. If you changed a section name, you may have been looking at the file for the old name.
- See the warning regarding changes to the HTML settings at section 7.6.
- Verify that the proper CSS is actually being used.
- The browser may compensate for some subtle changes, such as automatically generating ligatures, reflowing text, etc.

Un-matched conditional compiles:

- Verify the proper `begin`/`end` of `warpprint`, `warpHTML`, and `warpall` environments.

13.2.1 Debug tracing output

`\tracinglwarp` When `\tracinglwarp` is used, lwarp will add extra tracing messages to the `.log` file. The last several messages may help track down errors.

Place `\tracinglwarp` just after `\usepackage{lwarp}` to activate tracing.

13.3 Compiling the `lwarp.dtx` file

`lwarp_tutorial.tex`: Copy or link `lwarp_tutorial.txt` from the TDS doc directory to the `source` directory, or wherever you wish to compile the documentation. This file is included verbatim in the documentation, but is in the `doc` directory so that it may be found by *texdoc* and copied by the user.

Illogical error messages caused by an out-of-sync `lwarp.sty` file:

1. Delete the `lwarp.sty` file.
2. Enter **`pdflatex lwarp.ins`** to generate a new `lwarp.sty` file.
3. Enter **`pdflatex lwarp.dtx`** to recompile the `lwarp.pdf` documentation.

Un-nested environments:

Be sure to properly nest:

- `\begin{macrocode}` and `\end{macrocode}`
- `\begin{macro}` and `\end{macro}`
- `\begin{environment}` and `\end{environment}`

14 Trademarks

- TEX is a trademark of American Mathematical Society.
- ADOBE® and ADOBE *Framemaker*® are either registered trademarks or trademarks of ADOBE SYSTEMS INCORPORATED in the United States and/or other countries.
- LINUX® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- MAC OS® is a trademark of APPLE INC.
- MADCAP FLARE™ is the property of MADCAP SOFTWARE, INC.
- MATHJAX is copyright 2009 and later. The MATHJAX CONSORTIUM is a joint venture of the AMERICAN MATHEMATICAL SOCIETY (AMS) and the SOCIETY FOR INDUSTRIAL AND APPLIED MATHEMATICS (SIAM) to advance mathematical and scientific content on the web.
- MICROSOFT®, ENCARTA, MSN, and WINDOWS® are either registered trademarks or trademarks of MICROSOFT CORPORATION in the United States and/or other countries.
- UNIX® is a registered trademark of THE OPEN GROUP.
- YOUTUBE™ is trademark of GOOGLE LLC.

File 1 **lwarp.sty**

15 Implementation

This package is perhaps best described as a large collection of smaller individual technical challenges, in many cases solved through a number of ~~crude hacks~~ clever tricks. Reference sources are given for many of the solutions, and a quick internet search will provide additional possibilities.

Judgement calls were made, and are often commented. Improvements are possible. The author is open to ideas and suggestions.

Packages were patched for re-use where they provided significant functionality. Examples include xcolor with its color models and conversion to HTML color output, and siunitx which provides many number and unit-formatting options, almost all of which are available in pure-text form, and thus easily used by *pdftotext*.

Packages were emulated where their primary purpose was visual formatting which is not relevent to HTML output. For example, packages related to sectioning are already patched by numerous other packages, creating a difficult number of combinations to try to support, and yet in HTML output all of the formatting is thrown away, so these packages are merely emulated.

Packages with graphical output are allowed as-is, but must be nested inside a `latexmage` environment to preserve the graphics.

Testing has primarily been done with the Iceweasel/Firefox browser.

Table 12: Section depths and HTML headings

Section	LaTeX depth	HTML headings *
Title of the entire website		`<h1>`
(none)	-5	new for this package
book	-2	`<div class = "book">`
part	-1	`<h2>`
chapter	0	`<h3>`
section	1	`<h4>`
subsection	2	`<h5>`
subsubsection	3	`<h6>`
paragraph	4	`<span class = "paragraph">`
subparagraph	5	`<span class = "subparagraph">`
listitem	7	new for this package, used for list items

* If `FormatWP` is true, section headings may be adjusted, depending on `WPTitleHeading`. See table 11 on page 185.

16 Section depths and HTML headings

Stacks are created to track depth inside the LaTeX document structure. This depth is translated to HTML headings as shown in table 12. "Depth" here is not depth in the traditional computer-science stack-usage sense, but rather a representation of the nesting depth inside the LaTeX document structure.

When starting a new section, the program first must close out any existing sections and lists of a deeper level to keep the HTML tags nested correctly.

Support for the memoir package will require the addition of a `book` level, which may push the HTML headings down a step, and also cause `subsubsection` to become a `<div>` due to a limit of six HTML headings.

It is possible to use HTML5 `<section>` and `<h1>` for all levels, but this may not be well-recognized by older browsers.

Fixed levels for parts and chapters allow the CSS to remain fixed as well.

17 Source code

This is where the documented source code for lwarp begins, continuing through the following sections all the way to the change log and index at the end of this document.

The following sections document the actual implementation of the lwarp package.

line numbers — The small numbers at the left end of a line refer to line numbers in the `lwarp.sty` file.

subjects — Blue-colored tags in the left margin aid in quickly identifying the subject of each paragraph. These are often the targets of index entries.

Prog Lwarp

index entries — Black-colored tags in the left marign are used to identify programming objects such as files, packages, environments, booleans, and counters. Items without a tag are command macros. Each of these also appears in the index as individual entries, and are also listed together under “files”, “packages”, “environments”, “booleans”, and “counters”.

⚠ warnings — Special warnings are marked with a warning icon.

for HTML output:
for PRINT output:
for HTML & PRINT:

Green-colored tags in the left margin show which sections of source code apply to the generation of HTML, print, or both forms of output.

— lwarp source code begins on the following page —

18 Detecting the TEX engine — *pdflatex*, *lualatex*, *xelatex*

See: http://tex.stackexchange.com/a/47579.

Detects XETEX and LUALATEX:

```
\RequirePackage{iftex}[2019/11/07]
\RequirePackage{ifpdf}
\RequirePackage{ifptex}

\newif\ifxetexorluatex

\ifXeTeX
    \xetexorluatextrue
\else
    \ifLuaTeX
        \xetexorluatextrue
    \else
        \xetexorluatexfalse
    \fi
\fi
```

19 Early package requirements

etoolbox (*Pkg*) Provides \ifbool and other functions.

```
\RequirePackage{etoolbox}[2011/01/03]% v2.6 for \BeforeBeginEnvironment, etc.
```

Patch to fix copy of environment with a \par:

https://github.com/josephwright/etoolbox/issues/35

```
\long\def\etb@carsquare#1#2#3\@nil{#1#2}
```

xpatch (*Pkg*) Patches macros with optional arguments.

```
\RequirePackage{xpatch}
```

ifplatform (*Pkg*) Provides \ifwindows to try to automatically detect WINDOWS OS.

```
\RequirePackage{ifplatform}% sense op-system platform
```

letltxmacro (*Pkg*)

```
\RequirePackage{letltxmacro}
```

20 Package load order

Several packages must never be used with lwarp, others should only be loaded before lwarp, and others should only be loaded after. The lwarp core checks most

of these cases. In some lwarp-* packages, \LWR@loadbefore is used to trigger an error if they are loaded after lwarp, while additional code provides necessary patches for when they are loaded before.

Packages which must be loaded after lwarp are enfoced by a large number of \LWR@loadafter statements, below. Some packages are emulated by memoir, and so these are tested by \LWR@notmemoirloadafter, which does not cause an error if memoir is used.

\LWR@checkloadfilename is used to check each filename to see if it must never be loaded, or must always be loaded before lwarp.

20.1 Tests of package load order

\LWR@loadafter {⟨*packagename*⟩} Error if this package was loaded before lwarp.

```
\newcommand*{\LWR@loadafter}[1]{%
\IfPackageLoadedTF{#1}
{
    \PackageError{lwarp}
        {%
            Package #1,\MessageBreak
            or one which uses #1,\MessageBreak
            must be loaded after Lwarp.\MessageBreak
            Enter 'H' for possible solutions%
        }
        {%
            Move ``\protect\usepackage{#1}'' after
            ``\protect\usepackage{lwarp}''.\MessageBreak
            Package #1 may also be loaded by something else,\MessageBreak
            which must also be moved after Lwarp.%
        }
}
{\relax}
}
```

\LWR@notmemoirloadafter {⟨*packagename*⟩} Error if not memoir class and this package was loaded before lwarp.

memoir emulates many packages, and pretends that they have already been loaded.

```
\IfClassLoadedTF{memoir}
{\newcommand*{\LWR@notmemoirloadafter}[1]{}}
{\LetLtxMacro\LWR@notmemoirloadafter\LWR@loadafter}
```

\LWR@notltjloadafter {⟨*packagename*⟩} Error if not a ltjs* class and this package was loaded before lwarp.

```
\LetLtxMacro\LWR@notltjloadafter\LWR@loadafter

\IfClassLoadedTF{ltjarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjsarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

```
\IfClassLoadedTF{ltjsbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjsreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjspf}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjskiyou}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjtarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjtbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\IfClassLoadedTF{ltjtreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

`\LWR@loadbefore` {⟨*packagename*⟩} Error if this package is loaded after lwarp.

```
\newcommand*{\LWR@loadbefore}[1]{%
\IfPackageLoadedTF{#1}
{\relax}
{
    \PackageError{lwarp}
    {%
        Package #1 must be loaded before lwarp.\MessageBreak
        Enter 'H' for possible solutions%
    }
   {Move ``\protect\usepackage{#1}'' before ``\protect\usepackage{lwarp}''.}
}
}
```

`\LWR@checkloadbefore` {⟨*packagename*⟩}

Given `\LWR@tempone` is the package name to compare to, if package names match, error if it is loaded after lwarp.

```
\newcommand*{\LWR@checkloadbefore}[1]{%
    \ifdefstring{\LWR@tempone}{#1}{%
        \LWR@loadbefore{#1}%
    }{}%
}
```

`\LWR@loadnever` {⟨*badpackagename*⟩} {⟨*replacementpkgnames*⟩}

The first packages is not supported, so tell the user to use the second instead. Factored from `\LWR@checkloadnever` and `\LWR@earlyloadnever`.

```
\newcommand*{\LWR@loadnever}[2]{%
\PackageError{lwarp}
{%
    Package #1 is not yet supported\MessageBreak
    by lwarp's HTML conversion%
    \ifblank{#2}{}{%
        .\MessageBreak
        Package(s)\MessageBreak
        \space\space#2\MessageBreak
        may be useful instead%
    }%
}
{%
    Package #1 might conflict with lwarp in some way,\MessageBreak
    or is superceded by another package.%
    \ifblank{#2}{}{%
        \MessageBreak
        For possible alternatives, see package(s) #2.%
```

```
    }%
}
}
```

`\LWR@afterloadnever` {⟨*badpackagename*⟩} {⟨*replacementpkgnames*⟩}

Given: `\LWR@tempone` is set to the package name being tested against, if this package name is the bad packagename, suggest the replacements instead. This is used when loading packages after lwarp.

```
\newcommand*{\LWR@afterloadnever}[2]{%
    \ifdefstring{\LWR@tempone}{#1}{%
        \LWR@loadnever{#1}{#2}%
    }{}%
}
```

`\LWR@earlyloadnever` {⟨*badpackagename*⟩} {⟨*replacementpkgname*⟩}

The first package is not supported, so tell the user to use the second instead. This version checks immediately for packages which may have been loaded before lwarp.

```
\newcommand*{\LWR@earlyloadnever}[2]{%
    \IfPackageLoadedTF{#1}{%
        \LWR@loadnever{#1}{#2}%
    }{}%
}
```

`\LWR@earlyclassloadnever` {⟨*badclassname*⟩} {⟨*replacementclassname*⟩}

The first class is not supported, so tell the user to use the second instead. This version checks immediately for classes which may have been loaded before lwarp.

```
\newcommand*{\LWR@earlyclassloadnever}[2]{%
\IfClassLoadedTF{#1}{%
\PackageError{lwarp}
{%
    Class #1 is not supported\MessageBreak
    by lwarp's HTML conversion%
    \ifblank{#2}{}{%
        .\MessageBreak
        #2 may be useful instead%
    }%
}
{%
    Class #1 might conflict with lwarp in some way,\MessageBreak
    or is superceded by another class.%
    \ifblank{#2}{}{%
        \MessageBreak
        For a possible alternative, see #2.%
    }%
}
}{\relax}%
}
```

20.2 Error for disallowed packages and classes loaded before lwarp

\LWR@checkloadnevers Checks against a list of incompatible packages.

```
\newcommand*{\LWR@checkloadnevers}{
\LWR@checkloadnever{ae}{cm-super, lmodern}
\LWR@checkloadnever{aecompl}{cm-super, lmodern}
\LWR@checkloadnever{aecc}{cm-super, lmodern}
\LWR@checkloadnever{alg}{algorithm2e, algorithmicx}
\LWR@checkloadnever{algorithmic}{algorithm2e, algorithmicx}
\LWR@checkloadnever{bitfield}{bytefield}
```

bxcjkatype is based on CJK:

```
\LWR@checkloadnever{bxcjkjatype}{upLaTeX, bxjsarticle, ujarticle, utarticle}
```

```
\LWR@checkloadnever{caption2}{caption}
% \LWR@checkloadnever{ccaption}{caption}% might be preloaded by memoir
\LWR@checkloadnever{colortab}{colortbl}
\LWR@checkloadnever{csvtools}{datatool}
\LWR@checkloadnever{doublespace}{setspace}
\LWR@checkloadnever{fancyheadings}{fancyhdr}
\LWR@checkloadnever{fncylab}{cleveref}
\LWR@checkloadnever{formula}{siunitx}
\LWR@checkloadnever{glossary}{glossaries}
```

hangul is not in TeXLive, and is not tested:

```
\LWR@checkloadnever{hangul}{kotex, xetexko, luatexko}
```

```
\LWR@checkloadnever{hyper}{hyperref}
\LWR@checkloadnever{libgreek}{libertinust1math, newtx}
\LWR@checkloadnever{newthm}{ntheorem}
\LWR@checkloadnever{pdfcprot}{microtype}
\LWR@checkloadnever{picins}{floatflt, wrapfig, wrapfig2}
\LWR@checkloadnever{rplain}{fancyhdr}
\LWR@checkloadnever{si}{siunitx}
\LWR@checkloadnever{sistyle}{siunitx}
\LWR@checkloadnever{slashbox}{diagbox}
\LWR@checkloadnever{statex}{statex2}
\LWR@checkloadnever{t1enc}{fontenc, inputenc, inputenx}
\LWR@checkloadnever{ucs}{inputenc, inputencx}
\LWR@checkloadnever{wasysym}{textcomp, amssymb, amsfonts, mnsymbol, fdsymbol}
```

The following may one day be supported by lwarp:

```
% \LWR@checkloadnever{adjustbox}{}% req'd for menukeys
\LWR@checkloadnever{animate}{}
\LWR@checkloadnever{auto-pst-pdf}{}
\LWR@checkloadnever{auto-pst-pdf-lua}{}
\LWR@checkloadnever{algorithms}{}
\LWR@checkloadnever{arraycols}{}
\LWR@checkloadnever{bidi}{}
\LWR@checkloadnever{cals}{}
```

```
\LWR@checkloadnever{cellspace}{tabls}
```

```
\LWR@checkloadnever{cgloss4e}{}
\LWR@checkloadnever{collcell}{}
\LWR@checkloadnever{colophon}{}
\LWR@checkloadnever{cooltooltips}{}
\LWR@checkloadnever{covington}{}
\LWR@checkloadnever{crbox}{}
\LWR@checkloadnever{decision-table}{}
\LWR@checkloadnever{dvgloss}{}
\LWR@checkloadnever{ednotes}{}
\LWR@checkloadnever{edfnotes}{}
\LWR@checkloadnever{eledform}{}
\LWR@checkloadnever{eledmac}{}
\LWR@checkloadnever{embedfile}{}
\LWR@checkloadnever{endnotes-hy}{endnotes}
\LWR@checkloadnever{expex}{}
\LWR@checkloadnever{fancytooltips}{}
\LWR@checkloadnever{fixocgx}{}
\LWR@checkloadnever{flowfram}{}
\LWR@checkloadnever{gb4e}{}
\LWR@checkloadnever{gmverse}{}
\LWR@checkloadnever{graphbox}{}
\LWR@checkloadnever{graphicxbox}{}
\LWR@checkloadnever{hvfloat}{}
\LWR@checkloadnever{inline-images}{}
\LWR@checkloadnever{isorot}{rotating}
\LWR@checkloadnever{ledmac}{}
\LWR@checkloadnever{linguex}{}
\LWR@checkloadnever{longdiv}{}
\LWR@checkloadnever{longfigure}{}
\LWR@checkloadnever{longtabu}{}
\LWR@checkloadnever{mdwenv}{}
\LWR@checkloadnever{mdwlist}{}
\LWR@checkloadnever{mdwtab}{}
\LWR@checkloadnever{navigator}{}
\LWR@checkloadnever{nccpic}{}
\LWR@checkloadnever{nccsect}{}
\LWR@checkloadnever{newvbtm}{}
\LWR@checkloadnever{ocg-p}{}
\LWR@checkloadnever{ocgtools}{}
\LWR@checkloadnever{ocgx}{}
\LWR@checkloadnever{ocgx2}{}
\LWR@checkloadnever{parrun}{}
\LWR@checkloadnever{poemscol}{}
\LWR@checkloadnever{poetry}{}
\LWR@checkloadnever{program}{}
\LWR@checkloadnever{proofread}{}
\LWR@checkloadnever{pst-pdf}{}
\LWR@checkloadnever{refstyle}{}
\LWR@checkloadnever{robustindex}{}
\LWR@checkloadnever{robustglossary}{}
\LWR@checkloadnever{semioneside}{}
\LWR@checkloadnever{slemph}{}
\LWR@checkloadnever{snotez}{sidenotes}
\LWR@checkloadnever{spacingtricks}{}
\LWR@checkloadnever{sverb}{verbatim, fancyvrb}
\LWR@checkloadnever{syntax}{}
\LWR@checkloadnever{tablists}{}
\LWR@checkloadnever{tabto}{}
\LWR@checkloadnever{tabu}{}
\LWR@checkloadnever{tabularht}{}
```

```
\LWR@checkloadnever{tabularkv}{}
\LWR@checkloadnever{thumby}{}
\LWR@checkloadnever{titles}{}
\LWR@checkloadnever{typehtml}{}
\LWR@checkloadnever{unicode-bidi}{}
\LWR@checkloadnever{vcell}{}
\LWR@checkloadnever{xhfill}{}
}
```

`\LWR@checkloadnever` {⟨*badpackagename*⟩} {⟨*replacementpkgname*⟩}

The first package is not supported, so tell the user to use the second instead.

When lwarp is first loaded, this is set to `\LWR@earlyloadnever` to check for incompatible packages which were loaded before lwarp. After lwarp is loaded, this is changed to `\LWR@afterloadnever` to check for incompatible packages during `\usepackage`.

```
\LetLtxMacro\LWR@checkloadnever\LWR@earlyloadnever
```

Now check for incompatible packages which have been loaded before lwarp:

```
\LWR@checkloadnevers
```

The older CJK and CJKutf8 only work with xeCJK:

```
\IfPackageLoadedTF{xeCJK}{}{
    \LWR@checkloadnever{CJK}{ctex, xeCJK}
    \LWR@checkloadnever{CJKutf8}{ctex, xeCJK}
}
```

Some classes do not work with lwarp:

```
\LWR@earlyclassloadnever{beamer}{beamerarticle}
\LWR@earlyclassloadnever{jarticle}{ujarticle}
\LWR@earlyclassloadnever{jbook}{ujbook}
\LWR@earlyclassloadnever{jreport}{ujreport}
\LWR@earlyclassloadnever{tarticle}{utarticle}
\LWR@earlyclassloadnever{tbook}{utbook}
\LWR@earlyclassloadnever{treport}{utreport}
\LWR@earlyclassloadnever{novel}{}
\LWR@earlyclassloadnever{powerdot}{}
```

20.3 Enforcing package loading after lwarp

Packages which should only be loaded after lwarp are tested here to trip an error of they have already been loaded.

The following packages must be loaded after lwarp:

```
\LWR@loadafter{2in1}
\LWR@loadafter{2up}
\LWR@loadafter{a4}
\LWR@loadafter{a4wide}
\LWR@loadafter{a5comb}
\LWR@notmemoirloadafter{abstract}
```

```
\LWR@loadafter{academicons}
\LWR@loadafter{accents}
\LWR@loadafter{accessibility}
\LWR@loadafter{accsupp}
\LWR@loadafter{acro}
\LWR@loadafter{acronym}
\LWR@loadafter{adjmulticol}
\LWR@loadafter{addlines}
\LWR@loadafter{afterpage}
\LWR@loadafter{algorithm2e}
\LWR@loadafter{algorithmicx}
\LWR@loadafter{alltt}
\LWR@loadafter{amscdx}
% \LWR@loadafter{amsmath}% may be preloaded
% \LWR@loadafter{amsthm}% may be preloaded
\LWR@loadafter{anonchap}
\LWR@loadafter{anysize}
\LWR@notmemoirloadafter{appendix}
\LWR@loadafter{ar}
\LWR@loadafter{arabicfront}
\LWR@notmemoirloadafter{array}
\LWR@loadafter{arydshln}
\LWR@loadafter{asymptote}
% \LWR@loadafter{atbegshi}% now in LaTeX core, also used by morewrites
\LWR@loadafter{attachfile}
\LWR@loadafter{attachfile2}
\LWR@loadafter{authblk}
\LWR@loadafter{authoraftertitle}% Supported as-is, but must be loaded after.
\LWR@loadafter{autobreak}
\LWR@loadafter{autonum}
\LWR@loadafter{awesomebox}
\LWR@loadafter{axessibility}
\LWR@loadafter{axodraw2}
\LWR@loadafter{backnaur}
\LWR@loadafter{backref}
\LWR@loadafter{balance}
\LWR@loadafter{bbding}
\LWR@loadafter{beamerarticle}
\LWR@loadafter{bigdelim}
\LWR@loadafter{bigfoot}
\LWR@loadafter{bigstrut}
\LWR@loadafter{bitpattern}
\LWR@loadafter{blowup}
\LWR@loadafter{bm}
\LWR@loadafter{booklet}
\LWR@loadafter{bookmark}
\LWR@notmemoirloadafter{booktabs}
\LWR@loadafter{bophook}
\LWR@loadafter{bounddvi}
\LWR@loadafter{boxedminipage}
\LWR@loadafter{boxedminipage2e}
\LWR@loadafter{braket}
\LWR@loadafter{breakurl}
\LWR@loadafter{breqn}
\LWR@loadafter{bsheaders}
\LWR@loadafter{bussproofs}
\LWR@loadafter{bxpapersize}
\LWR@loadafter{bytefield}
\LWR@loadafter{ccicons}
\LWR@loadafter{cancel}
```

```
\LWR@loadafter{canoniclayout}
\LWR@loadafter{caption}
\LWR@loadafter{caption2}
\LWR@loadafter{caption3}
\LWR@loadafter{cases}
% catoptions is supported by the lwarp core
% \LWR@loadafter{ccaption}% may be preloaded by memoir
\LWR@loadafter{centerlastline}
% \LWR@loadafter{centernot}% may be preloaded by newtx
\LWR@loadafter{changebar}
\LWR@loadafter{changelayout}
\LWR@notmemoirloadafter{changepage}
\LWR@loadafter{changes}
\LWR@loadafter{chappg}
\LWR@loadafter{chapterbib}
\LWR@loadafter{chemfig}
\LWR@loadafter{chemformula}
\LWR@loadafter{chemgreek}
\LWR@loadafter{chemmacros}
\LWR@loadafter{chemnum}
\LWR@loadafter{chkfloat}
\LWR@notmemoirloadafter{chngpage}
\LWR@loadafter{cite}
\LWR@loadafter{citeref}
\LWR@loadafter{classicthesis}
\LWR@loadafter{cleveref}
% cmbright may be preloaded
\LWR@loadafter{cmdtrack}
\LWR@loadafter{colonequals}
\LWR@loadafter{color}
\LWR@loadafter{colortbl}
\LWR@loadafter{continue}
\LWR@loadafter{copyrightbox}
\LWR@notmemoirloadafter{crop}
% ctex must be loaded before lwarp
\LWR@loadafter{ctable}
\LWR@loadafter{cuted}
\LWR@loadafter{cutwin}
\LWR@loadafter{dblfloatfix}
\LWR@loadafter{dblfnote}
\LWR@notmemoirloadafter{dcolumn}
\LWR@loadafter{decimal}
\LWR@loadafter{decorule}
\LWR@loadafter{diagbox}
\LWR@loadafter{dingbat}
\LWR@loadafter{DotArrow}
\LWR@loadafter{dotlessi}
\LWR@loadafter{dprogress}
\LWR@loadafter{draftcopy}
\LWR@loadafter{draftfigure}
\LWR@loadafter{draftwatermark}
\LWR@loadafter{drftcite}
\LWR@loadafter{easy-todo}
\LWR@loadafter{ebook}
\LWR@loadafter{econometrics}
\LWR@loadafter{ed}
\LWR@loadafter{ellipsis}
\LWR@loadafter{embrac}
\LWR@loadafter{emptypage}
\LWR@loadafter{endfloat}
```

```
\LWR@loadafter{endheads}
\LWR@loadafter{endnotes}
\LWR@loadafter{engtlc}
\LWR@loadafter{enotez}
\LWR@notmemoirloadafter{enumerate}
\LWR@loadafter{enumitem}
\LWR@notmemoirloadafter{epigraph}
\LWR@loadafter{epsf}
\LWR@loadafter{epsfig}
\LWR@loadafter{epstopdf}
\LWR@loadafter{epstopdf-base}
\LWR@loadafter{eqlist}
\LWR@loadafter{eqparbox}
\LWR@loadafter{errata}
\LWR@loadafter{eso-pic}
\LWR@loadafter{esvect}
\LWR@loadafter{etoc}
\LWR@loadafter{eurosym}
\LWR@loadafter{everypage}
% \LWR@loadafter{everyshi}% now in LaTeX core
\LWR@loadafter{extarrows}
\LWR@loadafter{extramarks}
\LWR@loadafter{fancybox}
\LWR@loadafter{fancyhdr}
\LWR@loadafter{fancypar}
\LWR@loadafter{fancyref}
\LWR@loadafter{fancytabs}
\LWR@loadafter{fancyvrb}
\LWR@loadafter{fbox}
\LWR@loadafter{fewerfloatpages}
\LWR@loadafter{figcaps}
\LWR@loadafter{figsize}
\LWR@loadafter{fitbox}
\LWR@loadafter{fix2col}
\LWR@loadafter{fixmath}
\LWR@loadafter{fixme}
\LWR@loadafter{fixmetodonotes}
\LWR@loadafter{flafter}
\LWR@loadafter{flippdf}
\LWR@loadafter{float}
\LWR@loadafter{floatflt}
\LWR@loadafter{floatpag}
\LWR@loadafter{floatrow}
\LWR@loadafter{fltrace}
\LWR@loadafter{flushend}
\LWR@loadafter{fnbreak}
\LWR@loadafter{fncychap}
\LWR@loadafter{fnlineno}
\LWR@loadafter{fnpara}
\LWR@loadafter{fnpos}
\LWR@loadafter{fontawesome}
\LWR@loadafter{fontawesome5}
% fontenc must be loaded before lwarp
% fontspec must be loaded before lwarp
\LWR@loadafter{footmisc}
\LWR@loadafter{footnote}
\LWR@loadafter{footnotebackref}
\LWR@loadafter{footnotehyper}
\LWR@loadafter{footnoterange}
\LWR@loadafter{footnpag}
```

```
\LWR@loadafter{foreign}
\LWR@loadafter{forest}
\LWR@loadafter{fouridx}
% fourier may be loaded before lwarp
\LWR@loadafter{framed}
\LWR@loadafter{froufrou}
\LWR@loadafter{ftcap}
\LWR@loadafter{ftnright}
\LWR@loadafter{fullminipage}
\LWR@loadafter{fullpage}
\LWR@loadafter{fullwidth}
\LWR@loadafter{fvextra}
\LWR@loadafter{fwlw}
\LWR@loadafter{gensymb}
\LWR@loadafter{gentombow}
% geometry is always loaded by lwarp, and lwarp-geometry is AtBeginDocument
\LWR@loadafter{ghsystem}
\LWR@loadafter{gindex}
\LWR@loadafter{glossaries}
\LWR@loadafter{gmeometric}
% \LWR@loadafter{graphics}% pre-loaded by xunicode
% \LWR@loadafter{graphicx}% pre-loaded by xunicode
\LWR@loadafter{gloss}
\LWR@loadafter{glossary}
\LWR@loadafter{grffile}
\LWR@loadafter{grid}
\LWR@loadafter{grid-system}
\LWR@loadafter{gridset}
\LWR@loadafter{hang}
\LWR@loadafter{hanging}
\LWR@loadafter{hepunits}
\LWR@loadafter{hhline}
\LWR@loadafter{hhtensor}
\LWR@loadafter{hypbmsec}
\LWR@loadafter{hypcap}
\LWR@loadafter{hypdestopt}
\LWR@loadafter{hypernat}
\LWR@loadafter{hyperref}
\LWR@loadafter{hyperxmp}
\LWR@loadafter{hyphenat}
\LWR@loadafter{idxlayout}
\LWR@loadafter{ifoddpage}
\LWR@loadafter{imakeidx}
\LWR@loadafter{impnattypo}
\LWR@notmemoirloadafter{index}
% inputenc must be loaded before lwarp
% inputenx must be loaded before lwarp
% inputtrc may be loaded before lwarp
\LWR@loadafter{intopdf}
\LWR@loadafter{isomath}
\LWR@loadafter{isotope}
\LWR@loadafter{jurabib}
\LWR@loadafter{karnaugh-map}
\LWR@loadafter{keyfloat}
\LWR@loadafter{keystroke}
% kpfonts may be loaded before lwarp
% kpfonts-otf may be loaded before lwarp
\LWR@loadafter{layaureo}
\LWR@loadafter{layout}
\LWR@loadafter{layouts}
```

```
\LWR@loadafter{leading}
\LWR@loadafter{leftidx}
\LWR@loadafter{letterspace}
\LWR@loadafter{lettrine}
% libertinust1math may be loaded before lwarp
\LWR@loadafter{lineno}
\LWR@loadafter{lips}
\LWR@loadafter{listings}
\LWR@loadafter{listliketab}
\LWR@loadafter{lltjp-siunitx}
\LWR@loadafter{lltjp-tascmac}
\LWR@loadafter{longtable}
\LWR@loadafter{lpic}
\LWR@loadafter{lscape}
\LWR@loadafter{ltablex}
\LWR@loadafter{ltcaption}
\LWR@loadafter{ltxgrid}
\LWR@loadafter{ltxtable}
\LWR@loadafter{lua-check-hyphen}
\LWR@loadafter{lua-visual-debug}
\LWR@loadafter{luacolor}
\LWR@loadafter{luamplib}
\LWR@loadafter{luatodonotes}
\LWR@loadafter{luavlna}
\LWR@loadafter{lyluatex}
\LWR@loadafter{magaz}
\LWR@notmemoirloadafter{makeidx}
\LWR@loadafter{manyfoot}
\LWR@loadafter{marginfit}
\LWR@loadafter{marginfix}
\LWR@loadafter{marginnote}
\LWR@loadafter{marvosym}
% mathalpha may be loaded before lwarp
\LWR@loadafter{mathastext}
\LWR@loadafter{mathcomp}
\LWR@loadafter{mathdesign}
\LWR@loadafter{mathdots}
\LWR@loadafter{mathfixs}
\LWR@loadafter{mathpazo}
\LWR@loadafter{mathptmx}
\LWR@loadafter{mathspec}
\LWR@loadafter{mathtools}
\LWR@loadafter{mattens}
\LWR@loadafter{maybemath}
\LWR@loadafter{mcaption}
\LWR@loadafter{mdframed}
\LWR@loadafter{mdwmath}
\LWR@loadafter{media9}
\LWR@loadafter{memhfixc}
\LWR@loadafter{menukeys}
\LWR@loadafter{metalogo}
\LWR@loadafter{metalogox}
\LWR@loadafter{mhchem}
\LWR@loadafter{microtype}
\LWR@loadafter{midfloat}
\LWR@loadafter{midpage}
\LWR@loadafter{minibox}
\LWR@loadafter{minitoc}
\LWR@loadafter{minted}
\LWR@loadafter{mismath}
```

```
\LWR@loadafter{mleftright}
% morefloats must be allowed early for print mode
\LWR@notmemoirloadafter{moreverb}
% morewrites must be loaded before lwarp
\LWR@notmemoirloadafter{movie15}
\LWR@notmemoirloadafter{mparhack}
\LWR@loadafter{multibib}
\LWR@loadafter{multicap}
%\LWR@loadafter{multicol}% loaded by ltxdoc
\LWR@loadafter{multicolrule}
\LWR@loadafter{multimedia}
\LWR@loadafter{multiobjective}
\LWR@loadafter{multirow}
\LWR@loadafter{multitoc}
\LWR@loadafter{musicography}
\LWR@loadafter{mwe}
\LWR@loadafter{nameauth}
\LWR@loadafter{nameref}
\LWR@loadafter{natbib}
\LWR@notmemoirloadafter{nccfancyhdr}
\LWR@loadafter{nccfoots}
\LWR@loadafter{nccmath}
\LWR@notmemoirloadafter{needspace}
% newclude must be loaded before lwarp
% newpxmath may be preloaded
% newtxmath may be loaded before lwarp
% newtxsf may be loaded before lwarp
% newunicodechar must be loaded before lwarp
\LWR@notmemoirloadafter{nextpage}
\LWR@loadafter{nicefrac}
\LWR@loadafter{niceframe}
\LWR@loadafter{nicematrix}
\LWR@loadafter{noitcrul}
\LWR@loadafter{nolbreaks}
\LWR@loadafter{nomencl}
\LWR@loadafter{nonfloat}
\LWR@loadafter{nonumonpart}
\LWR@loadafter{nopageno}
\LWR@loadafter{notes}
\LWR@loadafter{notespages}
\LWR@loadafter{nowidow}
\LWR@loadafter{ntheorem}
\LWR@loadafter{octave}
\LWR@loadafter{orcidlink}
\LWR@loadafter{overpic}
\LWR@loadafter{pagegrid}
\LWR@notmemoirloadafter{pagenote}
\LWR@loadafter{pagesel}
\LWR@loadafter{paralist}
\LWR@loadafter{parallel}
\LWR@loadafter{parcolumns}
\LWR@loadafter{parnotes}
\LWR@notmemoirloadafter{parskip}
\LWR@loadafter{pbalance}
\LWR@loadafter{pbox}
\LWR@loadafter{pdfcol}
\LWR@loadafter{pdfcolfoot}
\LWR@loadafter{pdfcolmk}
\LWR@loadafter{pdfcolparallel}
\LWR@loadafter{pdfcolparcolumns}
```

```
\LWR@loadafter{pdfcomment}
\LWR@loadafter{pdfcrypt}
\LWR@loadafter{pdflscape}
\LWR@loadafter{pdfmarginpar}
\LWR@loadafter{pdfpages}
\LWR@loadafter{pdfprivacy}
\LWR@loadafter{pdfrender}
\LWR@loadafter{pdfsync}
\LWR@loadafter{pdftricks}
\LWR@loadafter{pdfx}
\LWR@loadafter{perpage}
\LWR@loadafter{pfnote}
\LWR@loadafter{phfqit}
\LWR@loadafter{physics}
\LWR@loadafter{physunits}
\LWR@loadafter{picinpar}
\LWR@loadafter{pifont}
\LWR@loadafter{pinlabel}
\LWR@loadafter{placeins}
\LWR@loadafter{plarray}
\LWR@loadafter{plarydshln}
\LWR@loadafter{plextarray}
\LWR@loadafter{plextarydshln}
\LWR@loadafter{plcolortbl}
\LWR@loadafter{plextdelarray}
\LWR@loadafter{plimsoll}
\LWR@loadafter{prelim2e}
\LWR@loadafter{prettyref}
\LWR@loadafter{preview}
\LWR@loadafter{psfrag}
\LWR@loadafter{psfragx}
\LWR@loadafter{pst-eps}
\LWR@loadafter{pstool}
\LWR@loadafter{pstricks}
% \LWR@loadafter{pxatbegshi}% may be used by morewrites
\LWR@loadafter{pxeveryshi}
% \LWR@loadafter{pxfonts}% may be loaded before lwarp
\LWR@loadafter{pxftnright}
\LWR@loadafter{pxjahyper}
\LWR@loadafter{quotchap}
\LWR@loadafter{quoting}
\LWR@loadafter{ragged2e}
\LWR@loadafter{refcheck}
\LWR@loadafter{register}
\LWR@loadafter{relsize}
\LWR@loadafter{repeatindex}
\LWR@loadafter{resizegather}
\LWR@loadafter{returntogrid}
\LWR@loadafter{rlepsf}
\LWR@loadafter{rmathbr}
\LWR@loadafter{rmpage}
\LWR@loadafter{romanbar}
\LWR@loadafter{romanbarpagenumber}
\LWR@loadafter{rotating}
\LWR@loadafter{rotfloat}
\LWR@loadafter{rviewport}
\LWR@loadafter{savetrees}
% scalefnt is loaded by babel-french
\LWR@loadafter{scalerel}
\LWR@loadafter{schemata}
```

```
\LWR@loadafter{scrextend}
\LWR@loadafter{scrhack}
\LWR@loadafter{scrlayer}
\LWR@loadafter{scrlayer-notecolumn}
\LWR@loadafter{scrlayer-scrpage}
\LWR@loadafter{scrpage2}
\LWR@loadafter{section}
\LWR@loadafter{sectionbreak}
\LWR@loadafter{sectsty}
\LWR@loadafter{selectp}
\LWR@loadafter{semantic-markup}
\LWR@notmemoirloadafter{setspace}
\LWR@loadafter{shadow}
\LWR@loadafter{shapepar}
\LWR@notmemoirloadafter{showidx}
\LWR@loadafter{showlabels}
\LWR@loadafter{showkeys}
\LWR@loadafter{showtags}
\LWR@loadafter{shuffle}
\LWR@loadafter{sidecap}
\LWR@loadafter{sidenotes}
\LWR@loadafter{simplebnf}
\LWR@loadafter{SIunits}
\LWR@loadafter{siunitx}
\LWR@loadafter{siunitx-v2}
\LWR@loadafter{skmath}
\LWR@loadafter{slantsc}
\LWR@loadafter{slashed}
\LWR@loadafter{soul}
\LWR@loadafter{soulpos}
\LWR@loadafter{soulutf8}
\LWR@loadafter{splitbib}
\LWR@loadafter{splitidx}
\LWR@loadafter{srcltx}
\LWR@loadafter{srctex}
\LWR@loadafter{stabular}
\LWR@loadafter{stackengine}
\LWR@loadafter{stackrel}
\LWR@loadafter{statex2}
\LWR@loadafter{statistics}
\LWR@loadafter{statmath}
\LWR@loadafter{steinmetz}
\LWR@notltjloadafter{stfloats}
\LWR@loadafter{struktex}
\LWR@loadafter{subcaption}
\LWR@loadafter{subfig}
\LWR@loadafter{subfigure}
\LWR@loadafter{subsupscripts}
\LWR@loadafter{supertabular}
\LWR@loadafter{svg}
\LWR@loadafter{swfigure}
\LWR@loadafter{sympytex}
\LWR@loadafter{syntonly}
\LWR@loadafter{t1inc}
\LWR@loadafter{tabfigures}
\LWR@loadafter{tabls}
\LWR@loadafter{tablefootnote}
\LWR@notmemoirloadafter{tabularx}
\LWR@loadafter{tabulary}
\LWR@loadafter{tagpdf}
```

```
\LWR@loadafter{tagpdf-mc-code-generic}
\LWR@loadafter{tagpdf-mc-code-lua}
\LWR@loadafter{tascmac}
\LWR@loadafter{tcolorbox}
\LWR@loadafter{tensor}
\LWR@loadafter{termcal}
\LWR@loadafter{textarea}
% \LWR@loadafter{textcomp}% maybe before lwarp with font packages
\LWR@loadafter{textfit}
\LWR@loadafter{textpos}
\LWR@loadafter{theorem}
\LWR@loadafter{thinsp}
\LWR@loadafter{thm-listof}
\LWR@loadafter{thm-restate}
\LWR@loadafter{thmbox}
\LWR@loadafter{thmtools}
\LWR@loadafter{threadcol}
\LWR@loadafter{threeparttable}
\LWR@loadafter{threeparttablex}
\LWR@loadafter{thumb}
\LWR@loadafter{thumbs}
\LWR@loadafter{tikz}
\LWR@loadafter{tikz-imagelabels}
\LWR@loadafter{titleps}
\LWR@loadafter{titlesec}
\LWR@loadafter{titletoc}
\LWR@notmemoirloadafter{titling}
% \LWR@loadafter{tocbasic}% preloaded by koma-script classes
\LWR@notmemoirloadafter{tocbibind}
\LWR@loadafter{tocdata}
\LWR@loadafter{tocenter}
\LWR@notmemoirloadafter{tocloft}
\LWR@loadafter{tocstyle}
\LWR@loadafter{todo}
\LWR@loadafter{todonotes}
\LWR@loadafter{topcapt}
\LWR@loadafter{tram}
\LWR@loadafter{transparent}
\LWR@loadafter{trimclip}
\LWR@loadafter{trivfloat}
\LWR@loadafter{truncate}
\LWR@loadafter{turnthepage}
\LWR@loadafter{twoup}
% \LWR@loadafter{txfonts}% may be loaded before lwarp
% txgreeks may be loaded before lwarp

% \LWR@loadafter{typearea}% preloaded by koma-script classes
\LWR@loadafter{typicons}
% \LWR@loadafter{ulem}% preloaded by ctexart and related classes
\LWR@loadafter{umoline}
\LWR@loadafter{underscore}
% unicode-math may be loaded before lwarp
\LWR@loadafter{units}
\LWR@loadafter{unitsdef}
\LWR@loadafter{upgreek}
\LWR@loadafter{upref}
\LWR@loadafter{url}
\LWR@loadafter{ushort}
\LWR@loadafter{uspace}
\LWR@loadafter{varioref}
```

```
\LWR@notmemoirloadafter{verse}
\LWR@loadafter{versonotes}
\LWR@loadafter{vertbars}
\LWR@loadafter{vmargin}
\LWR@loadafter{vowel}
\LWR@loadafter{vpe}
\LWR@loadafter{vwcol}
\LWR@loadafter{wallpaper}
\LWR@loadafter{watermark}
\LWR@loadafter{widetable}
\LWR@loadafter{widows-and-orphans}
\LWR@loadafter{witharrows}
\LWR@loadafter{wrapfig}
\LWR@loadafter{wrapfig2}
\LWR@loadafter{xbmks}
\LWR@loadafter{xcolor}
\LWR@loadafter{xechangebar}
\LWR@loadafter{xellipsis}
% xetexko must be loaded before lwarp
\LWR@loadafter{xevlna}
\LWR@loadafter{xfakebold}
\LWR@loadafter{xfrac}
\LWR@loadafter{xltabular}
\LWR@loadafter{xltxtra}
\LWR@loadafter{xmpincl}
\LWR@loadafter{xpiano}
\LWR@loadafter{xpinyin}
\LWR@loadafter{xr}
\LWR@loadafter{xr-hyper}
\LWR@loadafter{xtab}
% xunicode must be loaded before lwarp
\LWR@loadafter{xurl}
\LWR@loadafter{xy}
\LWR@loadafter{zwpagelayout}
```

21 MD5 hashing

The MD5 hash is used for `lateximage` filenames for SVG math.

```
\newcommand{\LWR@mdfive}[1]{%
    \PackageError{lwarp}
        {No MD5 macro was found}
        {%
            Lwarp must find the macros \protect\pdfmdfivesum\space
            or \protect\mdfivesum.%
        }
}
```

The default for PDF LaTeX, DVI LaTeX, upLaTeX, etc:

```
\let\LWR@mdfive\pdfmdfivesum
```

For LuaLaTeX:

```
\ifLuaTeX
\RequirePackage{pdftexcmds}
```

```
\let\LWR@mdfive\pdf@mdfivesum
\fi
```

For XeLaTeX:

```
\ifXeTeX
\@ifundefined{pdffivesum}{}
    {\let\LWR@mdfive\pdfmdfivesum}
\@ifundefined{mdfivesum}{}
    {\let\LWR@mdfive\mdfivesum}
\fi
```

22 pdf LaTeX T1 and utf-8 encoding

When using pdf LaTeX, lwarp requires T1 font encoding, and recommends utf-8 input encoding.

If some other input encoding is already defined, lwarp will try to use it instead, and hope for the best.

XeLaTeX and LuaLaTeX are both utf-8 by nature.

`\LWR@pdfencoding` Sets `T1`, and also `utf8` if not already set.

```
\newcommand*{\LWR@pdfencoding}{%
    \RequirePackage[T1]{fontenc}

    \IfPackageLoadedTF{inputenc}{}{
        \IfPackageLoadedTF{inputenx}{}{
            \RequirePackage[utf8]{inputenc}
        }
    }
}

\ifPDFTeX% pdflatex or dvi latex
    \LWR@pdfencoding
\fi

\ifpTeX
    \LWR@pdfencoding
\fi
```

23 Unicode input characters

for HTML & PRINT: If using *pdflatex*, convert a minimal set of Unicode characters. Additional characters may be defined by the user, as needed.

A commonly-used multiply symbol is declared to be `\texttimes`.

The first arguments of `\newunicodechar` below are text ligatures in the source code, even though they are not printed in the following listing.

```
\ifpTeX
```

```
\else
\RequirePackage{newunicodechar}

\newunicodechar{×}{\texttimes}

\ifPDFTeX% pdflatex or dvi latex
\newunicodechar{ff}{ff}% Here, the first arguments are ligatures.
\newunicodechar{fi}{fi}
\newunicodechar{fl}{fl}
\newunicodechar{ffi}{ffi}
\newunicodechar{ffl}{ffl}
\newunicodechar{–}{---}
\newunicodechar{-}{--}
\fi

\fi
```

24 Avoid a bitmapped font

If DVI or PDF LaTeX, and if the default Computer Modern is the selected font family, ensure that cm-super or lmodern is used to provide a vector font.

```
\ifxetexorluatex
\else
    \ifdefstring{\f@family}{cmr}{
        \IfFileExists{type1ec.sty}% found in cm-super
        {}
        {% cm-super not installed
            \IfFileExists{lmodern.sty}{
                \PackageInfo{lwarp}{cm-super not installed, loading lmodern}
                \RequirePackage{lmodern}
            }{
                \PackageError{lwarp}
                {%
                    Lwarp requires a vector font.\MessageBreak
                  Install and load cm-super, lmodern, or another\MessageBreak
                    Type-1 vector font before loading lwarp.\MessageBreak
                    Enter 'H' for possible solutions%
                }
                {%
                    Install cm-super or lmodern.\MessageBreak
                    If lmodern, load it before lwarp:\MessageBreak
                       \space\space\protect\usepackage{lmodern}\MessageBreak
                         \space\space\protect\usepackage{lwarp}%
                }
            }
        }% cm-super not installed
    }{}% f@family
\fi
```

25 Upright quotes

In PDF TeX, preserve upright quotes in verbatim text. upquote also loads textcomp.

```
\ifPDFTeX
\RequirePackage{upquote}
\fi

\ifpTeX
    \RequirePackage{upquote}
\fi
```

26 Avoid bad font combinations

For XeLaTeX and LuaLaTeX, certain font combinations cause problems with lwarp.

libertinus-otf has special handling for `\textquotedbl`. Search for `\LWR@orig@textquotedbl`.

```
\ifxetexorluatex
    \AtBeginDocument{
        \IfPackageLoadedTF{kpfonts}{
            \PackageError{lwarp}
                {%
                    When using XeLaTeX or LuaLaTeX,\MessageBreak
                    use kpfonts-otf instead of kpfonts%
                }
                {%
                    Replace: \protect\usepackage{kpfonts}\MessageBreak
                    with: \protect\usepackage{kpfonts-otf}
                }
        }{}
    }
\fi
```

27 Miscellaneous tools

27.1 Variables

```
\newlength{\LWR@templengthone}
\newlength{\LWR@templengthtwo}
\newlength{\LWR@templengththree}
\newcounter{LWR@tempcountone}
```

27.2 Lengths and units

`\LWR@providelength` {⟨`\lengthname`⟩} Provides the length if it isn't defined yet.

Used to provide source compatibility for lengths which will be ignored, but might or might not be already provided by other packages.

```
\newcommand*\LWR@providelength[1]{%
    \ifdeflength{#1}{}{\newlength{#1}}%
}
```

`\LWR@convertto` {⟨*dest unit*⟩} {⟨*length*⟩}

Prints a length in the given units, without printing the unit itself.

```
\newcommand*{\LWR@convertto}[2]{\strip@pt\dimexpr #2*65536/\number\dimexpr 1#1}
```

`\LWR@printpercentlength` {⟨*smaller*⟩} {⟨*larger*⟩}

Prints a percent ratio of the two lengths.

```
\newcommand*{\LWR@printpercentlength}[2]{%
    \setcounter{LWR@tempcountone}{100*\ratio{#1}{#2}}%
    \arabic{LWR@tempcountone}%
}
```

27.3 Counters

`\defaddtocounter` {⟨*name*⟩} {⟨*value*⟩}

Locally add to a counter.

```
\providecommand*{\defaddtocounter}[2]{%
    \defcounter{#1}{\value{#1}+#2}%
}
```

27.4 Patching macros

`\LWR@patcherror` {⟨*packagename*⟩} {⟨*macroname*⟩}

Prints an error if could not patch a macro.

```
\newcommand*{\LWR@patcherror}[2]{%
    \PackageError{lwarp}%
        {%
            Unable to patch package #1,\MessageBreak
            macro \LWRbackslash #2.\MessageBreak
            Lwarp or #1 may need to be updated%
        }%
        {Please contact the maintainer of the Lwarp package.}%
}
```

27.5 Copying macros

`\csNewCommandCopycs` {⟨*dest csname*⟩} {⟨*source csname*⟩}

Given a cs-name for each, copies a macro to a new definition.

```
\providecommand*{\csNewCommandCopycs}[2]{%
    \expandafter\NewCommandCopy\csname#1\expandafter\endcsname%
        \csname#2\endcsname%
}
```

`\NewEnvironmentCopy` {⟨*dest*⟩} {⟨*source*⟩}

Copies an environment to a new definition.

```
\providecommand*{\NewEnvironmentCopy}[2]{%
    \csNewCommandCopycs{#1}{#2}%
    \csNewCommandCopycs{end#1}{end#2}%
}
```

27.6 Chinese text isolation

`\LWR@isolate` {⟨*text*⟩} Isolates Chinese characters from the surrounding text. This is required to avoid extra spaces on either side of the Chinese characters, especially when written to a file.

```
\newcommand{\LWR@isolate}[1]{#1}%

\IfPackageLoadedTF{ctexpatch}{
    \renewcommand{\LWR@isolate}[1]{\null#1\null}%
}{}

\IfPackageLoadedTF{xeCJK}{
    \renewcommand{\LWR@isolate}[1]{\null#1\null}%
}{}
```

`\LWR@disablepinyin` Disable xpinyin during file, sideTOC, and footnote generation. Set by xpinyin.

```
\newcommand*{\LWR@disablepinyin}{}
```

27.7 Inserting vertical space

`\LWR@forceemptyline` Extra vertical space in the HTML output. Use after `\LWR@stoppars`.

```
\newcommand*{\LWR@forceemptyline}{%
    \LWR@origrule{0pt}{1\baselineskip}%
    \LWR@orignewline%
}
```

27.8 Argument selection

`\LWR@thirdofthree` {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩}

`\LWR@fourthoffour` {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩}

`\LWR@firstoffive` {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

`\LWR@secondoffive` {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

`\LWR@thirdoffive` {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

`\LWR@fourthoffive` {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

\LWR@fifthoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

Expands to the nth of the five arguments. Used for extra cross referencing.

```
\long\def\LWR@thirdofthree#1#2#3{#3}%
\long\def\LWR@fourthoffour#1#2#3#4{#4}%

\long\def\LWR@firstoffive#1#2#3#4#5{#1}
\long\def\LWR@secondoffive#1#2#3#4#5{#2}
\long\def\LWR@thirdoffive#1#2#3#4#5{#3}
\long\def\LWR@fourthoffive#1#2#3#4#5{#4}
\long\def\LWR@fifthoffive#1#2#3#4#5{#5}
```

27.9 Inside boxes

Greater than zero if currently inside a TEX box, thus should not use \LWR@orignewpage. See section 13.2.

```
\newcounter{LWR@texboxdepth}
\setcounter{LWR@texboxdepth}{0}
```

\LWR@maybe@orignewpage Only do \LWR@orignewpage if not inside a TEX box.

```
\newcommand*{\LWR@maybe@orignewpage}{%
    \LWR@traceinfo{LWR@maybe@orignewpage}%
    \ifnumgreater{\value{LWR@texboxdepth}}{0}
        {}%
        {\LWR@orignewpage}%
    \LWR@traceinfo{LWR@maybe@orignewpage done}%
}
```

27.10 Global boxes

\LWR@gsavebox {⟨*macroname*⟩} {⟨*contents*⟩}

From https://tex.stackexchange.com/questions/288702/savebox-forgets-its-content-across-columns-inside-align

```
\DeclareRobustCommand\LWR@gsavebox[1]{%
  \@ifnextchar(%)
    {\LWR@@gsavepicbox#1}{\@ifnextchar[{\LWR@@gsavebox#1}{\LWR@gsbox#1}}}%
\long\def\LWR@gsbox#1#2{\global\setbox#1\hbox{%
  \color@setgroup#2\color@endgroup}}
\def\LWR@@gsavebox#1[#2]{%
  \@ifnextchar [{\LWR@@igsavebox#1[#2]}{\LWR@@igsavebox#1[#2][c]}}
\long\def\LWR@@igsavebox#1[#2][#3]#4{%
  \LWR@gsbox#1{\@imakebox[#2][#3]{#4}}}
\def\LWR@@gsavepicbox#1(#2,#3){%
  \@ifnextchar[%]
    {\LWR@@igsavepicbox#1(#2,#3)}{\LWR@@igsavepicbox#1(#2,#3)[]}}
\long\def\LWR@@igsavepicbox#1(#2,#3)[#4]#5{%
  \LWR@gsbox#1{\@imakepicbox(#2,#3)[#4]{#5}}}
```

LWR@glrbox (*env.*) {⟨*macroname*⟩}

```
\def\LWR@glrbox#1{%
  \edef\reserved@a{%
    \endgroup
    \global\setbox#1\hbox{%
      \begingroup\aftergroup}%
        \def\noexpand\@currenvir{\@currenvir}%
        \def\noexpand\@currenvline{\on@line}}%
  \reserved@a
    \@endpefalse
    \color@setgroup
      \ignorespaces}
\let\LWR@endglrbox\LWR@endlrbox
```

27.11 Converting a macro name to a cs name

\macrotocsname {⟨*macro name with backslash*⟩}

Results in the macro name without the leading backslash.

Ref: https://tex.stackexchange.com/questions/42318/removing-a-backslash-from-a-character-sequence

```
\newcommand*{\macrotocsname}[1]{%
  \ifcat\relax\noexpand#1%
    \expandafter\expandafter\expandafter\@gobble\expandafter\string
  \fi
  #1%
}
```

27.12 Title case

\LWRtexttitlecase

```
\ExplSyntaxOn
\newcommand*{\LWRtexttitlecase}[1]{%
    \text_titlecase:n{#1}%
}
\ExplSyntaxOff
```

27.13 LetLtxMacrocs

\LWR@LetLtxMacrocs {⟨*newcsname*⟩} {⟨*oldcsname*⟩}

\LetLtxMacro with cs names.

```
\newcommand*{\LWR@LetLtxMacrocs}[2]{%
    \expandafter\LetLtxMacro\csname #1\expandafter\endcsname%
    \csname#2\endcsname%
}
```

27.14 Absorbing a star

`\LWR@absorbstar` {⟨*csname*⟩}

Modifies a macro to aborb a star. Used for cleveref, since hyperref is emulated, so the starred macros are not created by cleveref.

```
\newcommand*{\LWR@absorbstar}[1]{%
    \LWR@LetLtxMacrocs{LWR@origns@#1}{#1}%
    \csdef{#1}{\@ifstar{\csuse{LWR@origns@#1}}{\csuse{LWR@origns@#1}}}
    \expandafter\robustify\csname #1\endcsname
}
```

28 Operating-System portability

`Unix` (*Prog*)
`Mac OS` (*Prog*)
`Linux` (*Prog*)
`MS-Windows` (*Prog*)
`Windows` (*Prog*)
`OSWindows` (*Opt*)

lwarp tries to detect which operating system is being used. UNIX / MAC OS / LINUX is the default (collectively referred to as "UNIX" in the configuration files), and MS-WINDOWS is supported as well.

If MS-WINDOWS is not correctly detected, use the lwarp option `OSWindows`.

When detected or specified, the operating-system path separator used by lwarp is modified, and the boolean `usingOSWindows` is set `true`. This boolean may be tested by the user for later use.

28.1 Literal characters

Literal characters to be used in `PrintLatexCmd` and `HTMLLatexCmd`. These are defined without `@` to easily allow their inclusion in the user's document.

The literal `%` character:

```
\let\LWRpercent\@percentchar
```

The literal `$` character:

```
\catcode`\$=12
\def\LWRdollar{$}
\def\LWRdollar{$}% syntax highlighting
\catcode`\$=3
```

The literal `&` character:

```
\catcode`\&=12
\def\LWRamp{&}
\catcode`\&=4
```

The literal `\` character. The ampersand is temporarily set to the escape character during the definition of the backslash macro.

```
\catcode`\&=0
&catcode`&\=12
&def&LWRbackslash{\}
```

```
&catcode`&\=0
\catcode`\&=4
```

The literal { character. The ampersand is temporarily set to the begin group character during the definition of the leftbrace macro.

```
\catcode`\&=1
\catcode`\{=12
\def\LWRleftbrace&{}
\catcode`\{=1
\catcode`\&=4
```

The literal } character. The ampersand is temporarily set to the end group character during the definition of the leftbrace macro.

```
\catcode`\&=2
\catcode`\}=12
\def\LWRrightbrace{}&
\catcode`\}=2
\catcode`\&=4
```

The literal # character:

```
\catcode`\#=12
\def\LWRhash{#}
\catcode`\#=6
```

`\LWRopquote` The operating system's quote mark, UNIX default. For WINDOWS, see `\LWR@setOSWindows`, below.

```
\def\LWRopquote{'}
```

`\LWRopseq` The operating system's sequential execution command, UNIX default. For WINDOWS, see `\LWR@setOSWindows`, below.

```
\def\LWRopseq{\space\LWRamp\LWRamp\space\space}
```

28.2 Common portability code

`usingOSWindows` (*bool*) Set if the `OSWindows` option is used, or if WINDOWS is automatically detected.

```
\newbool{usingOSWindows}
\boolfalse{usingOSWindows}
```

28.3 UNIX, LINUX, and MAC OS

`\OSPathSymbol` Symbol used to separate directories in a path.

```
\newcommand*{\OSPathSymbol}{/}
```

28.4 MS-Windows

For MS-Windows:

`\LWR@setOSWindows` Set defaults for the MS-Windows operating system. lwarp attempts to auto-detect the operatings system, and the `OSWindows` option may also be used to force MS-Windows compatibility.

```
\newcommand*{\LWR@setOSWindows}
{
\booltrue{usingOSWindows}
\renewcommand*{\OSPathSymbol}{\@backslashchar}
\def\LWRopquote{"}
\def\LWRopseq{\space\LWRamp\space\space}
}
```

Test for windows during compile. The user may also specify `OSWindows` package option in case this test fails.

```
\ifwindows
\LWR@setOSWindows
\fi
```

29 Package options

`kvoptions` (*Pkg*) Allows key/value package options.

```
\RequirePackage{kvoptions}
\SetupKeyvalOptions{family=LWR,prefix=LWR@}
```

`\lwarpsetup` A user interface to set the keys:

```
\newcommand{\lwarpsetup}[1]{\setkeys{LWR}{#1}}
```

`warpingprint` (*bool*)
`warpingHTML` (*bool*)
`mathjax` (*bool*)
`LWR@origmathjax` (*bool*)

Set to true/false depending on the package option selections for print/HTML/EPUB output and mathsvg/mathjax.

`LWR@origmathjax` remembers the original setting to be restored by `\displaymathnormal`.

```
\newbool{warpingprint}
\newbool{warpingHTML}
\newbool{mathjax}
\newbool{LWR@origmathjax}
```

defaults The default is print output, and SVG math if the user chose HTML output.

```
\booltrue{warpingprint}%
\boolfalse{warpingHTML}%
\boolfalse{mathjax}%
```

warpdisable (*Opt*) If the warpdisable option is given, both boolean warpingprint and boolean warpingHTML are false, and may be used for \ifbool tests. This option may be used to disable almost all of lwarp, for testing purposes.

```
\DeclareVoidOption{warpdisable}{%
    \PackageInfo{lwarp}{Using option 'warpdisable'}
    \boolfalse{warpingprint}%
    \boolfalse{warpingHTML}%
}
```

warppprint (*Opt*) If the warppprint option is given, boolean warpingprint is true and boolean warpingHTML is false, and may be used for \ifbool tests.

```
\DeclareVoidOption{warppprint}{%
    \PackageInfo{lwarp}{Using option 'warppprint'}
    \booltrue{warpingprint}%
    \boolfalse{warpingHTML}%
}
```

warpHTML (*Opt*) Anything in the warpHTML environment will be generated for HTML output only.

warpHTML (*Opt*) If the warpHTML option is given, boolean warpingHTML is true and boolean warpingprint is false, and may be used for \ifbool tests.

```
\DeclareVoidOption{warpHTML}{%
    \PackageInfo{lwarp}{Using option 'warpHTML'}%
    \booltrue{warpingHTML}%
    \boolfalse{warpingprint}%
}
```

mathsvg (*Opt*) Option mathsvg selects SVG math display: If the mathsvg option is given, boolean mathjax is false, and may be used for \ifbool tests.

```
\DeclareVoidOption{mathsvg}{%
    \PackageInfo{lwarp}{Using option 'mathsvg'}
    \boolfalse{mathjax}%
    \boolfalse{LWR@origmathjax}%

}
```

mathjax (*Opt*) Option mathjax selects MathJax math display: If the mathjax option is given, boolean mathjax is true, may be used for \ifbool tests.

```
\DeclareVoidOption{mathjax}{%
    \PackageInfo{lwarp}{Using option 'mathjax'}
    \booltrue{mathjax}%
    \booltrue{LWR@origmathjax}%

}
```

BaseJobname (*Opt*) Option BaseJobname sets the \BaseJobname for this document.

Default: \jobname

This is the \jobname of the printed version, even if currently compiling the HTML version. I.e. this is the \jobname without _html appended. This is used to set \HomeHTMLFilename if the user did not provide one.

```
\DeclareStringOption[\jobname]{BaseJobname}
```

ImagesDirectory (*Opt*)
Default: \jobname-images

Option ImagesDirectory sets the name of the directory to use for the lateximage images.

```
\DeclareStringOption[\BaseJobname-images]{ImagesDirectory}
```

ImagesName (*Opt*)
Default: image-

Option ImagesName sets the prefix to use for the lateximage images.

```
\DeclareStringOption[image-]{ImagesName}
```

makeindexStyle (*Opt*)
Default: lwarp.ist

Selects a custom .ist file. A customized file should be based on lwarp.ist. See section 8.6.20.

```
\DeclareStringOption[lwarp.ist]{makeindexStyle}
```

xindyStyle (*Opt*)
Default: lwarp.xdy

Selects a custom .xdy file. A customized file should be based on lwarp.xdy. See section 8.6.21.

```
\DeclareStringOption[lwarp.xdy]{xindyStyle}
```

xindyLanguage (*Opt*)
Default: english

Sets the *xindy* language to be assigned in *lwarpmk*'s configuration files. This is then used by *lwarpmk* while processing the index and glossary.

```
\DeclareStringOption[english]{xindyLanguage}
```

xindyCodepage (*Opt*)
Default: utf8

Sets the *xindy* codepage to be assigned in *lwarpmk*'s configuration files. This is then used by *lwarpmk* while processing the index.

```
\DeclareStringOption[utf8]{xindyCodepage}
```

xindexConfig (*Opt*)
Default: <empty>

Selects a custom xindex-*.lua file. A customized file should be based on xindex-cfg.lua. See section 8.6.22.

```
\DeclareStringOption[]{xindexConfig}
```

pdftotextEnc (*Opt*)
Default: UTF-8

The option pdftotextEnc sets the encoding used by *pdftotext*. This is passed to *pdftotext* using its -enc option, and is used when converting LaTeX PDF output with HTML tags into a plain-text file with HTML tags.

```
\DeclareStringOption[UTF-8]{pdftotextEnc}
```

lwarpmk (*Opt*)

Tells lwarp to generate a local copy of *lwarpmk* called lwarpmk.lua. Useful for archiving for future use. This file may be made executable and acts just like *lwarpmk*.

If lwarpmk option, creates a local copy of lwarpmk.lua:

```
\newbool{LWR@creatinglwarpmk}
\boolfalse{LWR@creatinglwarpmk}

\DeclareVoidOption{lwarpmk}{
    \PackageInfo{lwarp}{Using option 'lwarpmk'}
    \booltrue{LWR@creatinglwarpmk}
}
```

`OSWindows` (*Opt*) Tells lwarp to use MS-WINDOWS compatibility. Auto-detection of the operating system is attempted, and this option is only necessary if the auto-detection fails. See the automatically-generated `lwarpmk.conf` file to find out whether the operating system was detected correctly.

```
\DeclareVoidOption{OSWindows}{
    \PackageInfo{lwarp}{Using option 'OSWindows'}
    \LWR@setOSWindows
}
```

`HomeHTMLFilename` (*Opt*) Default: `\BaseJobname` The filename of the homepage. The default is the jobname. This option is stored into `\LWR@HomeHTMLFilename`, and later transferred into `\HomeHTMLFilename` for internal use.

```
\DeclareStringOption[]{HomeHTMLFilename}
```

`HTMLFilename` (*Opt*) Default: `<empty>` The filename prefix of web pages after the homepage. The default is empty, no prefix. This option is stored into `\LWR@HTMLFilename`, and later transferred into `\HTMLFilename` for internal use.

```
\DeclareStringOption[]{HTMLFilename}
```

`PrintLatexCmd` (*Opt*) Default: `<automatic>` The shell commands to use to compile the print document.

```
\DeclareStringOption[]{PrintLatexCmd}
```

`HTMLLatexCmd` (*Opt*) Default: `<automatic>` The shell commands to use to compile the HTML document.

```
\DeclareStringOption[]{HTMLLatexCmd}
```

`PrintIndexCmd` (*Opt*) Default: `<empty>` The shell commands to use to compile the print indexes.

```
\DeclareStringOption[]{PrintIndexCmd}
```

`HTMLIndexCmd` (*Opt*) Default: `<empty>` The shell commands to use to compile the HTML indexes.

```
\DeclareStringOption[]{HTMLIndexCmd}
```

`LatexmkIndexCmd` (*Opt*) Default: `<empty>` The shell commands to by used by *latexmk* to compile the print indexes. Unlike `PrintIndexCmd` and `HTMLIndexCmd`, `LatexmkIndexCmd` does not include the filename, which will be provided by *latexmk*.

```
\DeclareStringOption[]{LatexmkIndexCmd}
```

`makeindex` (*Opt*) Tells lwarp to use *makeindex* for index generation. When `lwarpmk.conf` and `*.lwarpmkconf` are generated, `PrintIndexCmd` and `HTMLIndexCmd` will be set for *makeindex* with a single index file.

```
\DeclareBoolOption[false]{makeindex}
```

`xindy` (*Opt*) Tells lwarp to use *xindy* for index generation. When `lwarpmk.conf` and `*.lwarpmkconf` are generated, `PrintIndexCmd` and `HTMLIndexCmd` will be set for *xindy* with a single index file.

```
\DeclareBoolOption[false]{xindy}
```

`xindex` (*Opt*) Tells lwarp to use *xindex* for index generation. When `lwarpmk.conf` and `*.lwarpmkconf` are generated, `PrintIndexCmd` and `HTMLIndexCmd` will be set for *xindex* with a single index file.

```
\DeclareBoolOption[false]{xindex}
```

`IndexRef` (*Opt*) Tells lwarp how to display the index entries in HTMLoutput. See section 7.5.
Default: `cref`

```
\DeclareStringOption[cref]{IndexRef}
```

`GlossaryCmd` (*Opt*) The shell command to use to compile the glossary. The print or HTML version of the glossary filename will be appended to this command.
Default: `makeglossaries`

```
\DeclareStringOption[makeglossaries]{GlossaryCmd}
```

`latexmk` (*Opt*) Option `latexmk` tells *lwarpmk* to use *latexmk* when compiling documents.

```
\DeclareBoolOption[false]{latexmk}
```

`dvips` (*Opt*) Option `dvips` tells *lwarpmk* to use *dvips* when compiling DVI *latex* documents.

```
\DeclareBoolOption[false]{dvips}
```

`dvipdfm` (*Opt*) Option `dvipdfm` tells *lwarpmk* to use *dvipdfm* when compiling DVI *latex* documents.

```
\DeclareBoolOption[false]{dvipdfm}
```

`dvipdfmx` (*Opt*) Option `dvipdfmx` tells *lwarpmk* to use *dvipdfmx* when compiling DVI *latex* documents.

```
\DeclareBoolOption[false]{dvipdfmx}
```

Execute options Execute the package options, with the defaults which have been set just above:

```
\ProcessKeyvalOptions*\relax
```

29.1 Additional options support

Assign the `\BaseJobname` if the user hasn't provided one:

```
\providecommand*{\BaseJobname}{\LWR@BaseJobname}
```

Defaults unless already over-ridden by the user:

```
\ifcsempty{LWR@HomeHTMLFilename}{
    \newcommand*{\HomeHTMLFilename}{\BaseJobname}
}{
    \csedef{HomeHTMLFilename}{\LWR@HomeHTMLFilename}
}

\csedef{HTMLFilename}{\LWR@HTMLFilename}
```

Special handling for underscores in labels and filenames.

\LWR@sanitized The sanitized version of what was given to \LWR@sanitize. Characters are set to their detokenized versions. Required for underscores in labels and filenames.

```
\newcommand*{\LWR@sanitized}{}
```

\LWR@sanitize {⟨*text*⟩}

Sanitizes the text and returns the result in \LWR@sanitized.

```
\newcommand*{\LWR@sanitize}[1]{%
\edef\LWR@sanitized{#1}%
\edef\LWR@sanitized{\detokenize\expandafter{\LWR@sanitized}}%
}
```

Sanitize some string options to neutralize underscores.

```
\LWR@sanitize{\LWR@BaseJobname}
\edef\LWR@BaseJobname{\LWR@sanitized}

\LWR@sanitize{\LWR@ImagesDirectory}
\edef\LWR@ImagesDirectory{\LWR@sanitized}

\LWR@sanitize{\LWR@ImagesName}
\edef\LWR@ImagesName{\LWR@sanitized}
```

\LWR@PrintIndexCmd and \LWR@HTMLIndexCmd are tested to see if they are empty. If so, they are set to a reasonable defaults for a single index using *makeindex*, then possibly set to defaults for *xindy* if the lwarp xindy option was selected, then likewise for *xindex* if the xindex option was selected.

```
\ifdefempty{\LWR@PrintIndexCmd}{
    \renewcommand{\LWR@PrintIndexCmd}{%
        makeindex  -s \LWR@makeindexStyle \space  \jobname.idx%
    }
    \ifbool{LWR@xindy}{
        \renewcommand{\LWR@PrintIndexCmd}{%
            xindy
            -M \LWR@xindyStyle \space
            -L \LWR@xindyLanguage \space
            -C \LWR@xindyCodepage \space
            \jobname.idx%
        }
    }{}
    \ifbool{LWR@xindex}{
        \ifdefvoid{\LWR@xindexConfig}{
            \renewcommand{\LWR@PrintIndexCmd}{%
                xindex
                \jobname.idx%
            }
        }{
            \renewcommand{\LWR@PrintIndexCmd}{%
                xindex
                -c \LWR@xindexConfig \space
                \jobname.idx%
            }
        }
    }{}
}{}
```

```
1199
1200 \ifdefempty{\LWR@HTMLIndexCmd}{
1201     \renewcommand{\LWR@HTMLIndexCmd}{%
1202         makeindex  -s \LWR@makeindexStyle \space  \jobname_html.idx%
1203     }
1204     \ifbool{LWR@xindy}{
1205         \renewcommand{\LWR@HTMLIndexCmd}{%
1206             xindy
1207             -M \LWR@xindyStyle \space
1208             -L \LWR@xindyLanguage \space
1209             -C \LWR@xindyCodepage \space
1210             \jobname_html.idx%
1211         }
1212     }{}
1213     \ifbool{LWR@xindex}{
1214         \ifdefvoid{\LWR@xindexConfig}{
1215             \renewcommand{\LWR@HTMLIndexCmd}{%
1216                 xindex
1217                 \jobname_html.idx%
1218             }
1219         }{
1220             \renewcommand{\LWR@HTMLIndexCmd}{%
1221                 xindex
1222                 -c \LWR@xindexConfig \space
1223                 \jobname_html.idx%
1224             }
1225         }
1226     }{}
1227 }{}
1228
1229 \ifdefempty{\LWR@LatexmkIndexCmd}{
1230     \renewcommand{\LWR@LatexmkIndexCmd}{%
1231         makeindex  -s \LWR@makeindexStyle%
1232     }
1233     \ifbool{LWR@xindy}{
1234         \renewcommand{\LWR@LatexmkIndexCmd}{%
1235             xindy
1236             -M \LWR@xindyStyle \space
1237             -L \LWR@xindyLanguage \space
1238             -C \LWR@xindyCodepage%
1239         }
1240     }{}
1241     \ifbool{LWR@xindex}{
1242         \ifdefvoid{\LWR@xindexConfig}{
1243             \renewcommand{\LWR@LatexmkIndexCmd}{%
1244                 xindex
1245             }
1246         }{
1247             \renewcommand{\LWR@LatexmkIndexCmd}{%
1248                 xindex
1249                 -c \LWR@xindexConfig
1250             }
1251         }
1252     }{}
1253 }{}
```

29.2 Conditional compilation

`\warpprintonly` {⟨*contents*⟩}

Only process the contents if producing printed output.

```
\newcommand{\warpprintonly}[1]{\ifbool{warpingprint}{#1}{}}
```

`\warpHTMLonly` {⟨*contents*⟩}

Only process the contents if producing HTML output.

```
\newcommand{\warpHTMLonly}[1]{\ifbool{warpingHTML}{#1}{}}
```

`comment` (*Pkg*) Provides conditional code blocks.

Attempts to use versions or verbatim fail in some cases, and do not provide much of a speed benefit even when they do work.

```
\RequirePackage{comment}
```

`\LWR@includecomment` {⟨*env name*⟩} {⟨*partial filename*⟩}

`\LWR@excludecomment` {⟨*env name*⟩} {⟨*partial filename*⟩}

Use many comment cut files to avoid collision in case the user uses the comment package. Each filename is "`comment_#2.cut`". Based on the comment package.

```
\def\LWR@includecomment
 #1#2{\message{Lwarp: Including comment '#1'}%
    \csarg\def{After#1Comment}{%
        \CloseAndInputCutFile%
        \csundef{LWR@#1commentused}%
    }
    \csarg\def{#1}{%
        \endgroup
        \ifcsdef{LWR@#1commentused}{
            \PackageError{lwarp}%
                {Nested #1 environment}%
                {%
                    Environment #1 cannot be nested.\MessageBreak
                    This can happen when a package is loaded
                    from inside a\MessageBreak
                    #1 environment.%
                }%
        }{\relax}
        \csdef{LWR@#1commentused}{}
        \message{Including '#1' comment.}%
        \def\CommentCutFile{comment_#2.cut}
        \SetUpCutFile
        \ProcessComment{#1}
    }%
    \CommentEndDef{#1}
}

\def\LWR@excludecomment
```

```
#1#2{\message{Lwarp: Excluding comment '#1'}%
    \csarg\def{#1}{
        \endgroup
        \message{Excluding '#1' comment.}%
        \begingroup
           \def\CommentCutFile{comment_#2.cut}
            \def\ProcessCutFile{}%
           \def\ThisComment####1{}%
            \ProcessComment{#1}
    }%
    \csarg\def{After#1Comment}{\CloseAndInputCutFile \endgroup}
    \CommentEndDef{#1}}
```

warpall (*env.*) Anything in the warpall environment will be generated for print or HTML outputs.

```
\LWR@includecomment{warpall}{all}
```

warpHTML (*env.*) For HTML output:

```
\ifbool{warpingHTML}
    {\LWR@includecomment{warpHTML}{html}}
    {\LWR@excludecomment{warpHTML}{html}}
```

warpprint (*env.*) Anything in the warpprint environment will be generated for print output only.

```
\ifbool{warpingprint}
    {\LWR@includecomment{warpprint}{print}}
    {\LWR@excludecomment{warpprint}{print}}
```

If warpdisable, turn off both print and HTML output:

```
\ifboolexpr{bool {warpingprint} or bool {warpingHTML}}
    {}
    {
        \LWR@excludecomment{warpHTML}{html}
        \LWR@excludecomment{warpprint}{print}
        \LWR@excludecomment{warpMathJax}{mathjax}
    }
```

warpMathJax (*env.*) Only if MathJax is being used along with HTML.

```
\begin{warpprint}
\LWR@excludecomment{warpMathJax}{mathjax}
\end{warpprint}

\begin{warpHTML}
\ifbool{mathjax}
    {\LWR@includecomment{warpMathJax}{mathjax}}
    {\LWR@excludecomment{warpMathJax}{mathjax}}
\end{warpHTML}
```

warpsvg (*env.*) Only if SVG math is being used along with HTML, or in print mode.

```
\begin{warpprint}
\LWR@includecomment{warpsvg}{mathsvg}
```

```
\end{warpprint}

\begin{warpHTML}
\ifbool{mathjax}
    {\LWR@excludecomment{warpsvg}{mathsvg}}
    {\LWR@includecomment{warpsvg}{mathsvg}}
\end{warpHTML}
```

`LWRcreatelwarpmk` (*env.*) Optionally generate a local copy of *lwarpmk*. Default to no.

```
\ifbool{LWR@creatinglwarpmk}
    {\LWR@includecomment{LWRcreatelwarpmk}{lwarpmk}}
    {\LWR@excludecomment{LWRcreatelwarpmk}{lwarpmk}}
```

30 Required packages

These packages are automatically loaded by lwarp when generating HTML output. Some of them are also automatically loaded when generating print output, but some are not.

for HTML output:

```
\begin{warpHTML}
```

`fontspec` (*Pkg*) Load fontspec if necessary:

```
\ifxetexorluatex
\IfPackageLoadedTF{fontspec}{}{
    \usepackage[no-math]{fontspec}
}
```

The monospaced font is used for HTML tags, so turn off its TeX ligatures and common ligatures:

```
\defaultfontfeatures[\rmfamily]{Ligatures={NoCommon,TeX}}
\defaultfontfeatures[\sffamily]{Ligatures={NoCommon,TeX}}
\defaultfontfeatures[\ttfamily]{Ligatures=NoCommon}
\else
```

pdflatex only: Only pre-loaded if *pdflatex* is being used.

`microtype` (*Pkg*)

ligatures Older browsers don't display ligatures. Turn off letter ligatures, keeping LaTeX dash and quote ligatures, which may fail on older browers but at least won't corrupt written words.

```
\RequirePackage{microtype}

\microtypesetup{
    protrusion=false,
    expansion=false,
    tracking=false,
    kerning=false,
    spacing=false}
%    \begin{macrocode}
%
```

```
% Disable ligatures for typewriter fonts.
% The comma was causing issues with \MathJax\ and \cs{,} followed by a comma.
% Ligatures for f, q, t, etc used to be disabled for non-typewriter fonts, but
% are now allowed.
% \changes{v0.89}{2020/08/01}{Disable typewriter ligatures.}
% ^^A \DisableLigatures[{,},f,q,t,T,Q]{encoding = *,family = *}% previous
%    \begin{macrocode}
\DisableLigatures{encoding = *,family = tt*}

\fi

\end{warpHTML}
```

`geometry` (*Pkg*) Tactics to avoid unwanted page breaks and margin overflow:

- Uses a very long and wide page to minimize page breaks and margin overflow.
- Uses a scriptsize font.
- Uses extra space at the margin to avoid HTML tag overflow off the page.
- Forces a new PDF page before some environments.
- Forces line break between major pieces of long tags.

for HTML output:

```
\begin{warpHTML}
```

If geometry has not yet been loaded, use the preexising page and text sizes to be preserved for later reuse. These will be replaced by lwarp `\AtBeginDocument` with a very large page size to reduce HTML tag overflow off the page.

```
\IfPackageLoadedTF{geometry}
{}{
    \RequirePackage[
        reset,
        paperwidth=\paperwidth,
        paperheight=\paperheight,
        textwidth=\textwidth,
        textheight=\textheight,
        left=\oddsidemargin,
        top=\topmargin,
        marginparsep=\marginparsep,
        marginparwidth=\marginparwidth,
    ]{geometry}
}
```

Remember the original definitions for later reuse. If the geometry package is loaded by the user, lwarp-geometry will nullify the user-level originals.

```
\LetLtxMacro\LWR@origgeometry\geometry
\LetLtxMacro\LWR@orignewgeometry\newgeometry
\LetLtxMacro\LWR@origrestoregeometry\restoregeometry
\LetLtxMacro\LWR@origsavegeometry\savegeometry
\LetLtxMacro\LWR@origloadgeometry\loadgeometry
```

`LWR@allowanothergeometry` (*bool*) geometry may be loaded by the user before lwarp, after lwarp, or not at all. If before lwarp, it will have already been loaded by now and its page layout has

already been saved. If geometry is loaded after lwarp, its layout will be set at that time and the user macros nullified. `\AtEndPreamble` this layout will be saved. If the user never loads geometry, lwarp-geometry will be loaded `\AtBeginDocument`, but it should not change the page layout set here. This is controlled by the boolean `LWR@allowanothergeometry`. Geometry may be adjusted throughout the preamble until `\AtEndPreamble`, when this boolean is set `false`.

```
\newbool{LWR@allowanothergeometry}
\booltrue{LWR@allowanothergeometry}
```

Use `\AtEndPreamble` to avoid class and option conflict by changing settings after other packages load, instead of using geometry package options:

```
\AtEndPreamble{
```

Whatever geometry choices the user has made in the preamble, either before or after lwarp was loaded, are now saved for possible temporary reuse, such as by lyluatex.

See the lwarp-geometry section for what happens if geometry is loaded after lwarp.

```
\LWR@origsavegeometry{LWR@usergeometry}
```

The user's paper size is saved for later reuse, such as by the pdfpages or parallel packages.

```
\newlength{\LWR@userspaperwidth}
\setlength{\LWR@userspaperwidth}{\paperwidth}

\newlength{\LWR@userspaperheight}
\setlength{\LWR@userspaperheight}{\paperheight}

\newlength{\LWR@usersmarginparwidth}
\setlength{\LWR@usersmarginparwidth}{\marginparwidth}

\newlength{\LWR@userstextwidth}
\setlength{\LWR@userstextwidth}{\textwidth}

\newlength{\LWR@userstextheight}
\setlength{\LWR@userstextwidth}{\textheight}
```

For lwarp, use a very large page and margins to help avoid letting HTML tags run off the edge:

```
\LWR@origgeometry{
    reset,
    paperheight=190in,
    paperwidth=20in,
    left=2in,
    right=6in,
    top=1in,
    bottom=1in,
    heightrounded,%
}
```

The lwarp page geometry is saved for future restore:

```
\LWR@origsavegeometry{LWR@lwarpgeometry}
```

No longer adjust the page layout when lwarp-geometry is loaded `\AtBeginDocument`:

```
\boolfalse{LWR@allowanothergeometry}%
```

ltjsbook and other classes can print vertically, and require these to be reset by lwarp:

```
\setlength{\textheight}{0.8\paperheight}
\setlength{\textwidth}{0.7\paperwidth}

\@twosidefalse
\@mparswitchfalse
}% \AtEndPreamble

\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

`xparse` (*Pkg*)

LaTeX3 command argument parsing

```
\RequirePackage{xparse}
```

`calc` (*Pkg*)

```
\RequirePackage{calc}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`expl3` (*Pkg*)

LaTeX3 programming

```
\RequirePackage{expl3}
```

`gettitlestring` (*Pkg*)

Used to emulate `\nameref`.

```
\RequirePackage{gettitlestring}

\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

`filecontents` (*Pkg*)

Used to write helper files while creating the print version.

Recent versions of LaTeX (as of Fall 2019) now include the functionality of the filecontents package, but with a new optional argument used to specify whether to force the overwriting of an existing file. If an older LaTeX kernel is used, the original

filecontents package is used, but it is patched to throw away the new optional argument.

```
\@ifundefined{filec@ntents@opt}{% older kernel, discard optional args
    \RequirePackage{filecontents}
    \LetLtxMacro\LWR@orig@filec@ntents\filec@ntents
    \IfPackageAtLeastTF{filecontents}{2011/10/08}
    {
```

For a newer version of the filecontents package, simply discard the optional argument.

```
        \renewcommand*{\filec@ntents}[1][]{\LWR@orig@filec@ntents}
    }
    {% patch older package for morewrites
```

For an older version of filecontents, discard the optional argument, and also patch to work with morewrites, per https://tex.stackexchange.com/questions/312830/does-morewrites-not-support-filecontents-and-can-i-write-body-of-environment-us/312910

```
        \newwrite\fcwrite
        \renewcommand*{\filec@ntents}[1][]{%
            \def\chardef##1\write{\let\reserved@c\fcwrite}%
            \LWR@orig@filec@ntents%
        }
    }

}% older kernel
{% newer kernel
```

For a newer kernel with a `filecontents` environment which accepts the optional `overwrite` argument, use the environment as-is.

```
}% newer kernel, filecontents env accepts optional args, do not load package

\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`xifthen` (*Pkg*)

```
\RequirePackage{xifthen}
```

`verbatim` (*Pkg*)

```
\RequirePackage{verbatim}
```

`refcount` (*Pkg*)

Provides `\setcounterref`, `\setcounterpageref`, etc.

```
\RequirePackage{refcount}
```

`newfloat` (*Pkg*)

```
\RequirePackage{newfloat}
```

```
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

`xstring` (*Pkg*) ⚠ index

There was a short-term bug in xstring regarding `\IfInteger` which affected lwarp's index generation. The updated version is requested here.

```
\RequirePackage{xstring}[2019/02/01]
```

`environ` (*Pkg*)

Used to encapsulate math environments for re-use in HTML `<alt>` text.

```
\RequirePackage{environ}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`printlen` (*Pkg*)

Used to convert lengths for image width/height options.

```
\RequirePackage{printlen}
```

`\LWR@printlength` {⟨*length*⟩}

Prints a length using a locally-controlled unit and space. Rounding is used unless the length is small.

```
\newrobustcmd*{\LWR@printlength}[1]{%
    \begingroup%
    \uselengthunit{PT}%
    \renewcommand*{\unitspace}{}%
    \ifdimless{#1}{10pt}{%
        \printlength{#1}%
    }{%
        \rndprintlength{#1}%
    }%
    \endgroup%
}
```

```
\end{warpHTML}
```

31 Loading packages

`\RequirePackage` and `\usepackage` are modified to error-check for certain packages, and for HTML they load the lwarp- version if it exists.

for HTML & PRINT:

```
\begin{warpall}
```

Remember the original `\RequirePackage`:

```
\LetLtxMacro\LWR@origRequirePackage\RequirePackage
\LetLtxMacro\LWR@origRequirePackageWithOptions\RequirePackageWithOptions
```

\LWR@requirepackagenames Stores the list of required package names.

```
\newcommand*{\LWR@requirepackagenames}{}
```

\LWR@parsedrequirepackagenames Stores the parsed list of required package names after spaces are removed and `lwarp-` is prepended.

```
\newcommand*{\LWR@parsedrequirepackagenames}{}
```

\LWR@nullifycomment Remove the preexisting `comment` environment. Certain packages define it for their own use.

```
\newcommand*{\LWR@nullifycomment}{%
    \PackageInfo{lwarp}%
      {Nullifying the comment environment before loading \LWR@strresulttwo,}%
    \let\comment\relax%
    \let\endcomment\relax%
}
```

\LWR@findword `[`⟨*1: separator*⟩`]` {⟨*2: list*⟩} {⟨*3: index*⟩} `[`⟨*4: destination*⟩`]`

Note that argument 4 is passed directly to `\StrBetween`.

```
\newcommand*\LWR@findword[3][,]{%
    \StrBetween[#3,\numexpr#3+1]{#1#2#1}{#1}{#1}%
}
```

\LWR@checkloadnever {⟨*bad package name*⟩} {⟨*replacement package names*⟩}

From now on, check for incompatible packages loaded via `\usepackage`, instead of packages loaded before lwarp:

```
\LetLtxMacro\LWR@checkloadnever\LWR@afterloadnever
```

\LWR@checkloadfilename {⟨*filename*⟩} Checks if this filename should be loaded after lwarp, or never at all.

```
\newcommand*{\LWR@checkloadfilename}[1]{%
```

Remember the package name to compare with, to be used by `\LWR@checkloadnever` and `\LWR@checkloadbefore`.

```
    \edef\LWR@tempone{#1}%
```

Check against the list of packages which should never be loaded:

```
    \LWR@checkloadnevers
```

The following should only be loaded before lwarp:

```
    \LWR@checkloadbefore{ctex}
    \LWR@checkloadbefore{fontspec}
    \LWR@checkloadbefore{inputenc}
    \LWR@checkloadbefore{inputenx}
    \LWR@checkloadbefore{nfssext-cfr}
```

```
    \LWR@checkloadbefore{fontaxes}
    \LWR@checkloadbefore{kotex}
    \LWR@checkloadbefore{kpfonts}% textcomp option clash
    \LWR@checkloadbefore{luatexja}
    \LWR@checkloadbefore{luatexja-fontspec}
    \LWR@checkloadbefore{luatexko}
    \LWR@checkloadbefore{morewrites}
    \LWR@checkloadbefore{newclude}
    \LWR@checkloadbefore{newunicodechar}
    \LWR@checkloadbefore{plext}
    \LWR@checkloadbefore{xeCJK}
    \LWR@checkloadbefore{xetexko}
    \LWR@checkloadbefore{zxjatype}
}
```

`\LWR@lookforpackagename {⟨index⟩}`

If HTML, and if this is an lwarp-supported package name, re-direct it to the lwarp version by renaming it `lwarp-` followed by the original name.

Looks `index` deep into the list of package names, `\LWR@requirepackagenames`, and builds `\LWR@parsedrequirepackagenames` which is the modified list of names.

```
\newcommand*{\LWR@lookforpackagename}[1]{%
```

Find the `index`'th package name from the list:

```
\LWR@findword{\LWR@requirepackagenames}{#1}[\LWR@strresult]%
```

Remove blanks. The original name with blanks is in `LWR@strresult` and the final name with no blanks goes into `LWR@strresulttwo`.

```
\StrSubstitute[100]{\LWR@strresult}{ }{}[\LWR@strresulttwo]%
```

See if the package name was found:

```
\IfStrEq{\LWR@strresulttwo}{}%
{}% no filename
{% yes filename was found
```

Possible adjustments before loading the package. Maybe nullify the `comment` environment if the new package will be redefining it for a new purpose.

```
    \ifdefstring{\LWR@strresulttwo}{easyReview}{\LWR@nullifycomment}{}%
    \ifdefstring{\LWR@strresulttwo}{changes}{\LWR@nullifycomment}{}%
```

If HTML, check if the package should be loaded before lwarp, or never at all:

```
    \ifbool{warpingHTML}{\LWR@checkloadfilename{\LWR@strresulttwo}}{}%
```

If HTML, and if found, and if an lwarp-equivalent name exists, use `lwarp-*` instead.

```
    \ifboolexpr{
        bool{warpingHTML} and
        test{\IfFileExists{lwarp-\LWR@strresulttwo.sty}}
    }%
    {% lwarp-* file found
        \ifdefvoid{\LWR@parsedrequirepackagenames}{%
```

```
            \edef\LWR@parsedrequirepackagenames{lwarp-\LWR@strresulttwo}%
        }{%
            \edef\LWR@parsedrequirepackagenames{%
                \LWR@parsedrequirepackagenames,lwarp-\LWR@strresulttwo%
            }%
        }%
    }%
    {%
```

Otherwise, use the current package name.

```
        \ifdefvoid{\LWR@parsedrequirepackagenames}{%
            \edef\LWR@parsedrequirepackagenames{\LWR@strresulttwo}%
        }{%
            \edef\LWR@parsedrequirepackagenames{%
                \LWR@parsedrequirepackagenames,\LWR@strresulttwo%
            }%
        }%
    }% no lwarp-* file
}% yes filename
}
```

`\RequirePackage` `[`⟨*1: options*⟩`]` `{`⟨*2: package names*⟩`}` `[`⟨*3: version*⟩`]`

For each of many package names in a comma-separated list, if an lwarp version of a package exists, select it instead of the LaTeX version.

```
\RenewDocumentCommand{\RequirePackage}{o m o}{%
```

Redirect up to twenty names:[17]

```
\renewcommand*{\LWR@requirepackagenames}{#2}%
\renewcommand*{\LWR@parsedrequirepackagenames}{}%
\LWR@lookforpackagename{1}%
\LWR@lookforpackagename{2}%
\LWR@lookforpackagename{3}%
\LWR@lookforpackagename{4}%
\LWR@lookforpackagename{5}%
\LWR@lookforpackagename{6}%
\LWR@lookforpackagename{7}%
\LWR@lookforpackagename{8}%
\LWR@lookforpackagename{9}%
\LWR@lookforpackagename{10}%
\LWR@lookforpackagename{11}%
\LWR@lookforpackagename{12}%
\LWR@lookforpackagename{13}%
\LWR@lookforpackagename{14}%
\LWR@lookforpackagename{15}%
\LWR@lookforpackagename{16}%
\LWR@lookforpackagename{17}%
\LWR@lookforpackagename{18}%
\LWR@lookforpackagename{19}%
\LWR@lookforpackagename{20}%
```

Error if braces are used in optional argument. This can cause an error, so tell how to avoid.

[17] This was originally nine names, but then I came across a package which used twelve...

```
\IfSubStr{\detokenize\expandafter{#1}}{\LWRleftbrace}%
    {%
        \PackageError{lwarp}{%
            You used:\MessageBreak
            \protect\usepackage[#1]{#2}\MessageBreak
            Braces in the package options will fail with Lwarp.\MessageBreak
            Instead, use:\MessageBreak
            \protect\PassOptionsToPackage{#1}{#2}\MessageBreak
            \protect\usepackage{#2}\MessageBreak
            near the line number given below.\MessageBreak
            Enter 'h' for more info%
        }%
        {%
           See the Lwarp manual troubleshooting index entry for\MessageBreak
            ``package, options with braces''%
        }%
    }%
    {}% no brace
```

\RequirePackage depending on the options and version:

```
\IfValueTF{#1}%
{% options given
    \IfValueTF{#3}% version given?
        {\LWR@origRequirePackage[#1]{\LWR@parsedrequirepackagenames}[#3]}%
        {\LWR@origRequirePackage[#1]{\LWR@parsedrequirepackagenames}}%
}%
{% no options given
    \IfValueTF{#3}% version given?
        {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}[#3]}%
        {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}}%
}%
}
\LetLtxMacro\usepackage\RequirePackage
\@onlypreamble\RequirePackage
\@onlypreamble\usepackage


\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

\LWR@ProvidesPackagePass {⟨*pkgname*⟩} [⟨*version*⟩]

Uses the original package, including options.

```
\NewDocumentCommand{\LWR@ProvidesPackagePass}{m o}{
    \PackageInfo{lwarp}{%
        Using package `#1',\MessageBreak
        and adding lwarp modifications, including options,\MessageBreak%
    }%
    \IfValueTF{#2}%
        {\ProvidesPackage{lwarp-#1}[#2]}%
        {\ProvidesPackage{lwarp-#1}}%
    \DeclareOption*{%
        \PassOptionsToPackage{\CurrentOption}{#1}%
    }%
    \ProcessOptions\relax%
```

If using catoptions, an error occurs if a package is loaded with an option then loaded again with no options. lwarp does this if a package is preloaded then later patched. To avoid an error while using catoptions, if a package has already been loaded, it is loaded again with its original options.

```
    \IfPackageLoadedTF{#1}{%
        \edef\LWR@tempone{\csuse{opt@#1.sty}}%
        \IfValueTF{#2}%
            {%
                \expandafter\LWR@origRequirePackage%
                    \expandafter[\LWR@tempone]{#1}[#2]%
            }%
            {%
                \expandafter\LWR@origRequirePackage%
                    \expandafter[\LWR@tempone]{#1}%
            }%
    }{%
        \IfValueTF{#2}%
            {\LWR@origRequirePackage{#1}[#2]}%
            {\LWR@origRequirePackage{#1}}%
    }%
```

In some cases, the following seems to be required to avoid an "unknown option" error, such as when loading xcolor with options.

```
    \DeclareOption*{}%
    \ProcessOptions\relax%
}
```

`\LWR@ProvidesPackageDropA` {⟨*name*⟩} {⟨*date or -NoValue-*⟩}

Declares the package. Factored for reuse.

```
\newcommand*{\LWR@ProvidesPackageDropA}[2]{%
    \PackageInfo{lwarp}{%
        Replacing package '#1' with the lwarp version,\MessageBreak
        and discarding options,%
    }%
    \IfValueTF{#2}
    {\ProvidesPackage{lwarp-#1}[#2]}
    {\ProvidesPackage{lwarp-#1}}
}
```

`\LWR@ProvidesPackageDropB` Nullifies then processes the options.

Seems to be required when options contain curly braces, which were causing "`Missing \begin{document}`".

```
\newcommand*{\LWR@ProvidesPackageDropB}{%
% \ProcessOptions\relax% original LaTeX code
\let\ds@\@empty%         from the original \ProcessOptions
\edef\@curroptions{}%    lwarp modification to \ProcessOptions
\@process@ptions\relax%  from the original \ProcessOptions
}
```

`\LWR@ProvidesPackageDrop` {⟨*pkgname*⟩} [⟨*version*⟩]

Ignores the original package and uses lwarp's version instead. Drops/discards all options.

```
\NewDocumentCommand{\LWR@ProvidesPackageDrop}{m o}{
```

Declare the package:

```
\LWR@ProvidesPackageDropA{#1}{#2}
```

Ignore all options:

```
\DeclareOption*{}
```

Process the options:

```
\LWR@ProvidesPackageDropB
}
```

```
\end{warpHTML}
```

32 File handles

Defines file handles for writes.

for HTML & PRINT:

```
\begin{warpall}
```

`\LWR@quickfile` For quick temporary use only. This is reused in several places.

```
\newwrite\LWR@quickfile%
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@lateximagesfile` For `<project>-images.txt`:

```
\newwrite\LWR@lateximagesfile
```

```
\end{warpHTML}
```

33 Include a file

During HTML output, `\include{<filename>}` causes the following to occur:

1. lwarp creates `<filename>_html_inc.tex` whose contents are:
   `\input <filename>.tex`
2. `<filename>_html_inc.tex` is then `\included` instead of `<filename>.tex`.
3. `<filename>_html_inc.aux` is automatically generated and used by LaTeX.

for HTML output:
```
\begin{warpHTML}
```

`\@include` {⟨*filename*⟩} Modified to load `_html_inc` files.

(Below, `\clearpage` caused missing text, and was changed to `\newpage`.)

```
\def\@include#1 {%
\immediate\openout\LWR@quickfile #1_html_inc.tex% lwarp
\immediate\write\LWR@quickfile{\string\input{#1.tex}}% lwarp
\immediate\closeout\LWR@quickfile% lwarp
\LWR@maybe@orignewpage% changed from clearpage
\if@filesw
    \immediate\write\@mainaux{\string\@input{#1_html_inc.aux}}% changed
\fi
\@tempswatrue
\if@partsw
    \@tempswafalse
    \edef\reserved@b{#1}%
    \@for\reserved@a:=\@partlist\do
    {\ifx\reserved@a\reserved@b\@tempswatrue\fi}%
\fi
\if@tempswa
    \let\@auxout\@partaux
    \if@filesw
        \immediate\openout\@partaux #1_html_inc.aux % changed
        \immediate\write\@partaux{\relax}%
    \fi
    \@input@{#1_html_inc.tex}% changed
    \LWR@maybe@orignewpage% changed from clearpage
    \@writeckpt{#1}%
    \if@filesw
        \immediate\closeout\@partaux
    \fi
\else
    \deadcycles\z@
    \@nameuse{cp@#1}%
\fi
\let\@auxout\@mainaux%
}
```

```
\end{warpHTML}
```

34 Copying a file

for HTML output:
```
\begin{warpHTML}
```

`\LWR@copyfile` {⟨*source filename*⟩} {⟨*destination filename*⟩}

Used to copy the `.toc` file to `.sidetoc` to re-print the TOC in the sideTOC navigation pane.

```
\newwrite\LWR@copyoutfile   % open the file to write to
\newread\LWR@copyinfile     % open the file to read from

\newcommand*{\LWR@copyfile}[2]{%
    \LWR@traceinfo{LWR@copyfile: copying #1 to #2}
```

```
1703
1704     \immediate\openout\LWR@copyoutfile=#2
1705     \openin\LWR@copyinfile=#1
1706     \begingroup\endlinechar=-1
1707     \makeatletter
1708
1709     \LWR@traceinfo{LWR@copyfile: about to loop}
1710
1711     \loop\unless\ifeof\LWR@copyinfile
1712         \LWR@traceinfo{LWR@copyfile: one line}
1713      \read\LWR@copyinfile to\LWR@fileline  % Read one line and store it into \LWR@fileline
1714 %   \LWR@fileline\par                    % print the content into the pdf
1715 % print the content:
1716      \immediate\write\LWR@copyoutfile{\unexpanded\expandafter{\LWR@fileline}}%
1717     \repeat
1718     \immediate\closeout\LWR@copyoutfile
1719     \LWR@traceinfo{LWR@copyfile: done}
1720     \endgroup
1721 }

1722 \end{warpHTML}
```

35 Debugging messages

HTML comments

To have the HTML output include additional HTML comments, such as which is closing, use

```
\booltrue{HTMLDebugComments}
```

debugging information

To have debug information written to the log, use

```
\tracinglwarp
```

for HTML & PRINT:

```
1723 \begin{warpall}
```

`LWR@tracinglwarp` (*bool*) True if tracing is turned on.

```
1724 \newbool{LWR@tracinglwarp}
```

`\tracinglwarp` Turns on the debug tracing messages.

```
1725 \newcommand{\tracinglwarp}{\booltrue{LWR@tracinglwarp}}
```

`\LWR@traceinfo {`⟨*text*⟩`}` If tracing is turned on, writes the text to the `.log` file.

```
1726 \newcommand{\LWR@traceinfo}[1]{%
1727 \ifbool{LWR@tracinglwarp}%
1728 {%
1729     \typeout{*** lwarp: #1}%
1730 }%
1731 {}%
1732 }
```

`HTMLDebugComments` (*bool*) Add comments in HTML about closing s, sections, etc.

Default: `false`

```
\newbool{HTMLDebugComments}
\boolfalse{HTMLDebugComments}
```

If `\tracinglwarp`, show where preamble hooks occur:

```
\AfterEndPreamble{
\LWR@traceinfo{AfterEndPreamble}
}

\AtBeginDocument{
\LWR@traceinfo{AtBeginDocument}
}

\end{warpall}
```

36 Defining print and HTML versions of macros and environments

The following refers to defining objects inside lwarp, and may also be of some use for package authors to adapt their packages for lwarp. The following is not for the user's document.

Many macros and environments must be provided as both print and HTML versions.

While generating the print version of a document, the original macros as defined by LaTeX and its packages are used as-is.

While generating the HTML version of a document, the original macro or environment is redefined to call a new HTML version or a copy of the original print version. The new HTML versions of macros and environments are used most of the time. Copies of the print versions are used inside a `lateximage` environment, which draws and remembers an image of the printed output, and also several other places.

The general structure for providing print and HTML versions of a macro or environment is as follows:

For a preexisting macro: An HTML version is provided with a special name, inside a `warpHTML` environment, then `\LWR@formatted` is used to redefine and patch various macros:

```
\begin{warpHTML}
\newcommand{\LWR@HTML@name}{...}

\LWR@formatted{name}
\end{warpHTML}
```

`\LWR@formatted{name}` copies the original print version to a new name `\LWR@print@<name>`, then redefines `\name` to use either the print or HTML version depending on which mode lwarp is using.

For a preexisiting environment: The process is similar. Note the use of `\LWR@formattedenv` instead of `\LWR@formatted`.

```
\begin{warpHTML}
\newenvironment{LWR@HTML@name}{...}{..}

\LWR@formattedenv{name}
\end{warpHTML}
```

For a new macro or environment: The print version is defined inside `warpall`, so that it can also be seen and modified by during HTML outut.

```
\begin{warpall}
\newcommand{\name}{...}% The print version.
\end{warpall}

\begin{warpHTML}
\newcommand{\LWR@HTML@name}{...}

\LWR@formatted{name}
\end{warpHTML}
```

Similar for an environment, using `\formattedenv`.

In general, `\LWR@formatted` or `\LWR@formattedenv` are placed inside a `warpHTML` environment, and while producing an HTML document they do the following:

- Macros are modified:
  1. The pre-existing print version `\name` is saved as `\LWR@print@<name>`, unless `\LWR@print@<name>` is already defined.
  2. The original `\name` is redefined to call either the print or HTML version depending on which format is in use at the moment, as set by `\LWR@formatting`, which is defined as either "`print`" or "`HTML`".
- When lwarp is producing a print document, the original definitions are used, as well as any new definitions defined in `warpall` above.
- When lwarp is generating HTML output, `\LWR@formatting` is set to "`HTML`", and `\name` is directed to `\LWR@HTML@<name>`. For an environment, `\endname` is directed to `\endLWR@HTML@<name>`.
- When lwarp is generating HTML output but enters a `lateximage` environment, or for some other reason needs to draw images using the original print defintions, `\LWR@formatting` is changed to "`print`" and `\name` is then redirected to `\LWR@print@<name>`, which was the original `\name`.
- Since the new `\name` does not process any arguments, they are processed by `\LWR@print@name` or `\LWR@HTML@name`.

Expandable versions are also provided as well. These usually are necessary for anything which could appear inside a `tabular`, without which a "`Misplaced \omit`" error may occur.

⚠ Misplaced `\omit` error

```
\LWR@expandableformatted
\LWR@expandableformattedenv
```

(Older versions of lwarp used \LetLtxMacro for everything, but this could fail when using macros defined by xparse. This older system is still in use for many definitions.)

Print or disabled versions:

for HTML & PRINT:

```
\begin{warpall}

\newcommand*{\LWR@formatted}[1]{}
\newcommand*{\LWR@expandableformatted}[1]{}
\newcommand*{\LWR@formattedenv}[1]{}
\newcommand*{\LWR@expandableformattedenv}[1]{}

\end{warpall}
```

for HTML output: HTML versions:

```
\begin{warpHTML}
```

\LWR@formatting Remembers if selected print/HTML formatting.

Used while \LWR@restoreorigformatting, such as in an lateximage. May be set to either "print" or "HTML".

```
\newcommand*{\LWR@formatting}{HTML}
```

\LWR@formatted@checkname {⟨*name*⟩}

```
\newcommand*{\LWR@formatted@checkname}[1]{%
    \ifcsundef{#1}{%
        \ifcsundef{LWR@print@#1}{%
            \PackageError{lwarp}
            {%
                \LWRbackslash#1 or \protect\LWR@print@#1\MessageBreak
                must be defined before using \protect\LWR@formatted, etc%
            }
            {Perhaps #1 is misspelled.}
        }{\relax}%
    }{\relax}%
    \ifcsundef{LWR@HTML@#1}{%
        \PackageError{lwarp}
        {%
            \protect\LWR@HTML@#1 must be defined
            before using \protect\LWR@formatted, etc%
        }
        {Perhaps #1 is misspelled.}
    }{\relax}%
}
```

\LWR@formatted@checkendname {⟨*name*⟩}

```
\newcommand*{\LWR@formatted@checkendname}[1]{%
    \ifcsundef{end#1}{%
        \ifcsundef{endLWR@print@#1}{%
            \PackageError{lwarp}
            {%
                \protect\end#1 or \protect\endLWR@print@#1\MessageBreak
```

```
                must be defined before using \protect\LWR@formatted, etc%
            }
            {Perhaps #1 is misspelled.}
        }{\relax}%
    }{\relax}%
    \ifcsundef{endLWR@HTML@#1}{%
        \PackageError{lwarp}
        {%
            \protect\endLWR@HTML@#1 must be defined
            before using \protect\LWR@formatted, etc%
        }
        {Perhaps #1 is misspelled.}
    }{\relax}%
}
```

`\LWR@formatted {⟨macroname⟩}` No backslash in the macro name.

If not yet defined, defines `\LWR@print@<name>` as the original print-mode `\<name>`. Also redefines `\<name>` to use `\LWR@<format>@<name>`, where <format> is set by `\LWR@formatting`, and is `print` or `HTML`.

```
\renewcommand*{\LWR@formatted}[1]{%
    \LWR@formatted@checkname{#1}%
    \ifcsundef{LWR@print@#1}{%
        \csNewCommandCopycs{LWR@print@#1}{#1}%
    }{}%
    \ifcsundef{#1}{%
        \expandafter\newrobustcmd\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }{%
        \expandafter\renewrobustcmd\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }%
}
```

`\LWR@expandableformatted {⟨macroname⟩}` No backslash in the macro name.

An expandable version of `\LWR@formatted`.

```
\renewcommand*{\LWR@expandableformatted}[1]{%
    \LWR@formatted@checkname{#1}%
    \ifcsundef{LWR@print@#1}{%
        \csNewCommandCopycs{LWR@print@#1}{#1}%
    }{}%
    \ifcsundef{#1}{%
        \expandafter\newcommand\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }{%
        \expandafter\renewcommand\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }%
}
```

`\LWR@formattedenv {⟨environmentname⟩}`

If not yet defined, defines the environment LWR@print@<name> as the original print-mode <name>. Also redefines the environment <name> to use environment LWR@<format>@<name>, where <format> is set by \LWR@formatting, and is print or HTML.

```
\renewcommand*{\LWR@formattedenv}[1]{%
    \LWR@formatted@checkname{#1}%
    \LWR@formatted@checkendname{#1}%
    \ifcsundef{LWR@print@#1}{%
        \NewEnvironmentCopy{LWR@print@#1}{#1}%
    }{}%
    \DeclareDocumentEnvironment{#1}{}%
    {%
        \@nameuse{LWR@\LWR@formatting @#1}%
    }%
    {%
        \@nameuse{endLWR@\LWR@formatting @#1}%
    }%
}
```

\LWR@expandableformattedenv {⟨*environmentname*⟩}

An expandable version of LWR@formattedenv.

```
\renewcommand*{\LWR@expandableformattedenv}[1]{%
    \LWR@formatted@checkname{#1}%
    \LWR@formatted@checkendname{#1}%
    \ifcsundef{LWR@print@#1}{%
        \NewEnvironmentCopy{LWR@print@#1}{#1}%
    }{}%
    \DeclareExpandableDocumentEnvironment{#1}{}%
    {%
        \@nameuse{LWR@\LWR@formatting @#1}%
    }%
    {%
        \@nameuse{endLWR@\LWR@formatting @#1}%
    }%
}
```

```
\end{warpHTML}
```

37 HTML-conversion output modifications

These booleans modify the HTML output in various ways to improve conversion to EPUB or word processor imports.

for HTML & PRINT:
```
\begin{warpall}
```

37.1 User-level controls

FormatEPUB (*bool*) Default: false

Changes HTML output for easy EPUB conversion via an external program. Removes per-file headers, footers, and nav. Adds footnotes per chapter/section.

```
\newbool{FormatEPUB}
\boolfalse{FormatEPUB}
```

`FormatWP` (*bool*)
Default: `false`

Changes HTML output for easier conversion by a word processor. Removes headers and nav, prints footnotes per section, and also forces single-file output and turns off HTML debug comments.

```
\newbool{FormatWP}
\boolfalse{FormatWP}
```

`WPMarkFloats` (*bool*)
Default: `false`

Adds

```
=== begin table ===
. . .
=== end ===
```

or

```
=== begin figure ===
. . .
=== end ===
```

around floats while formatting for word processors. This helps identify boundaries of floats to be manually converted to word-processor frames and captions. [18]

```
\newbool{WPMarkFloats}
\boolfalse{WPMarkFloats}
```

`WPMarkMinipages` (*bool*)
Default: `false`

Adds

```
=== begin minipage ===
. . .
=== end minipage ===
```

around minipages while formatting for word processors. This helps identify boundaries of minipages to be manually converted to word-processor frames.

```
\newbool{WPMarkMinipages}
\boolfalse{WPMarkMinipages}
```

`WPMarkTOC` (*bool*)
Default: `true`

While formatting for word processors, adds

```
=== table of contents ===
```

where the Table of Contents would have been. This helps identify where to insert the actual TOC.

If set `false`*, the actual toc is printed instead.*

```
\newbool{WPMarkTOC}
\booltrue{WPMarkTOC}
```

`WPMarkLOFT` (*bool*)
Default: `false`

While formatting for word processors, adds

```
=== list of figures ===    and/or
=== list of tables ===
```

[18] Perhaps some day word processors will have HTML import options for identifying `<figure>` and caption tags for figures and tables.

where each of these lists would have been. This helps identify where to insert the actual lists.

If set `false`*, the actual lists are printed instead.*

```
\newbool{WPMarkLOFT}
\boolfalse{WPMarkLOFT}
```

`WPMarkMath` (*bool*)
Default: `false`

While formatting for word processors, prints math as LaTeX code instead of creating SVG images or MathJax. This is useful for cut/paste into the *LibreOffice Writer TeXMaths* extension.

```
\newbool{WPMarkMath}
\boolfalse{WPMarkMath}
```

`WPTitleHeading` (*bool*)
Default: `false`

While formatting for word processors, `true` sets the document title to `<h1>`, which is expected for HTML documents, but also causes the lower-level section headings to start at **Heading 2** when imported into LibreOffice. Set to `false` to cause the title to be plain text, and the section headings to begin at **Heading 1**.

See table 11 on page 185.

```
\newbool{WPTitleHeading}
\boolfalse{WPTitleHeading}
```

```
\end{warpall}
```

37.2 Heading adjustments

If formatting the HTML for a word processor, adjust heading levels.

If `WPTitleHeading` is true, adjust so that `part` is **Heading 1**.

If `WPTitleHeading` is false, use `<h1>` for the title, and set `part` to **Heading 2**.

for HTML output:

```
\begin{warpHTML}
```

```
\AtBeginDocument{
\ifbool{FormatWP}{
\@ifundefined{chapter}{
\ifbool{WPTitleHeading}{% part and section starting at h2
\renewcommand*{\LWR@tagtitle}{h1}
\renewcommand*{\LWR@tagtitleend}{/h1}
\renewcommand*{\LWR@tagpart}{h2}
\renewcommand*{\LWR@tagpartend}{/h2}
\renewcommand*{\LWR@tagsection}{h3}
\renewcommand*{\LWR@tagsectionend}{/h3}
\renewcommand*{\LWR@tagsubsection}{h4}
\renewcommand*{\LWR@tagsubsectionend}{/h4}
\renewcommand*{\LWR@tagsubsubsection}{h5}
\renewcommand*{\LWR@tagsubsubsectionend}{/h5}
\renewcommand*{\LWR@tagparagraph}{h6}
\renewcommand*{\LWR@tagparagraphend}{/h6}
\renewcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
\renewcommand*{\LWR@tagsubparagraphend}{/span}
}% WPTitleHeading
{% not WPTitleHeading, part and section starting at h1
```

```
\renewcommand*{\LWR@tagtitle}{div class=\textquotedbl{}title\textquotedbl}
\renewcommand*{\LWR@tagtitleend}{/div}
\renewcommand*{\LWR@tagpart}{h1}
\renewcommand*{\LWR@tagpartend}{/h1}
\renewcommand*{\LWR@tagsection}{h2}
\renewcommand*{\LWR@tagsectionend}{/h2}
\renewcommand*{\LWR@tagsubsection}{h3}
\renewcommand*{\LWR@tagsubsectionend}{/h3}
\renewcommand*{\LWR@tagsubsubsection}{h4}
\renewcommand*{\LWR@tagsubsubsectionend}{/h4}
\renewcommand*{\LWR@tagparagraph}{h5}
\renewcommand*{\LWR@tagparagraphend}{/h5}
\renewcommand*{\LWR@tagsubparagraph}{h6}
\renewcommand*{\LWR@tagsubparagraphend}{/h6}
}% not WPTitleHeading
}% chapter undefined
{% chapter defined
\ifbool{WPTitleHeading}{}
{% not WPTitleHeading, part and chapter starting at h1
\renewcommand*{\LWR@tagtitle}{div class=\textquotedbl{}title\textquotedbl}
\renewcommand*{\LWR@tagtitleend}{/div}
\renewcommand*{\LWR@tagpart}{h1}
\renewcommand*{\LWR@tagpartend}{/h1}
\renewcommand*{\LWR@tagchapter}{h2}
\renewcommand*{\LWR@tagchapterend}{/h2}
\renewcommand*{\LWR@tagsection}{h3}
\renewcommand*{\LWR@tagsectionend}{/h3}
\renewcommand*{\LWR@tagsubsection}{h4}
\renewcommand*{\LWR@tagsubsectionend}{/h4}
\renewcommand*{\LWR@tagsubsubsection}{h5}
\renewcommand*{\LWR@tagsubsubsectionend}{/h5}
\renewcommand*{\LWR@tagparagraph}{h6}
\renewcommand*{\LWR@tagparagraphend}{/h6}
\renewcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
\renewcommand*{\LWR@tagsubparagraphend}{/span}
}% not WPTitleHeading
}% chapter defined
}{}% FormatWP
}% AtBeginDocument

\end{warpHTML}
```

38 Remembering original formatting macros

for HTML output:

```
\begin{warpHTML}
```

Remember original definitions of formatting commands. Will be changed to HTML commands for most uses. Will be temporarily restored to original meaning inside any `lateximage` environment and inside a `tabbing` environment. Also nullify unused commands.

Some packages redefine `\#`, which is used to generate HTML, so the original must be remembered here.

```
\chardef\LWR@origpound=`\#
```

```
\let\LWR@origcomma\,
\let\LWR@origtilde~
\let\LWR@orighfil\hfil
\let\LWR@orighss\hss
\let\LWR@origllap\llap
\let\LWR@origrlap\rlap
\let\LWR@orighfilneg\hfilneg
\let\LWR@orighspace\hspace

\let\LWR@origrule\rule

\let\LWR@origmedskip\medskip
\let\LWR@origbigskip\bigskip
```

libertinus-otf has too much kerning for `\textquotedbl`, causing an extra space.

```
\LetLtxMacro\LWR@orig@@textquotedbl\textquotedbl
\LetLtxMacro\LWR@orig@textquotedbl\LWR@orig@@textquotedbl

\AtEndPreamble{
\IfPackageLoadedTF{libertinus-otf}{
   \renewcommand{\LWR@orig@textquotedbl}{\LWR@orig@@textquotedbl\kern-.15em}
    \LetLtxMacro\textquotedbl\LWR@orig@textquotedbl
}{}
}
```

```
\LetLtxMacro\LWR@origttfamily\ttfamily

\LetLtxMacro\LWR@origem\em

\LetLtxMacro\LWR@orignormalfont\normalfont

\let\LWR@origonecolumn\onecolumn

\let\LWR@origsp\sp
\let\LWR@origsb\sb

\LetLtxMacro\LWR@origunderline\underline
```

```
\let\LWR@orignewpage\newpage

\let\LWR@origpagestyle\pagestyle
\let\LWR@origthispagestyle\thispagestyle
\LetLtxMacro\LWR@origpagenumbering\pagenumbering

\let\LWR@orignewline\newline

\AtBeginDocument{% in case packages change definition
\let\LWR@orig@trivlist\@trivlist
\let\LWR@origtrivlist\trivlist
\let\LWR@origendtrivlist\endtrivlist
\LetLtxMacro\LWR@origitem\item
\LetLtxMacro\LWR@origitemize\itemize
\LetLtxMacro\LWR@endorigitemize\enditemize
\LetLtxMacro\LWR@origenumerate\enumerate
\LetLtxMacro\LWR@endorigenumerate\endenumerate
\LetLtxMacro\LWR@origdescription\description
\LetLtxMacro\LWR@endorigdescription\enddescription
\let\LWR@orig@mklab\@mklab
```

```
\let\LWR@origmakelabel\makelabel
\let\LWR@orig@donoparitem\@donoparitem
\LetLtxMacro\LWR@orig@item\@item
\let\LWR@orig@nbitem\@nbitem
}

\let\LWR@origpar\par

\LetLtxMacro\LWR@origfootnote\footnote
\let\LWR@orig@mpfootnotetext\@mpfootnotetext


\AtBeginDocument{% in case packages change definition
\LetLtxMacro\LWR@orighline\hline%
\LetLtxMacro\LWR@origcline\cline%
}

\end{warpHTML}
```

39 Accents

Native LaTeX accents such as \" will work, but many more kinds of accents are available when using Unicode-aware XeLaTeX and LuaLaTeX. If using accents in section names which will become file names, it is recommended to use the LaTeX accents such as \" and \v instead of Unicode accents. The LaTeX accents will have the accents stripped when creating the filenames, whereas the Unicode accents will appear in the file names, which may cause issues with some operating systems.

for HTML output:

```
\begin{warpHTML}
```

Without `\AtBeginDocument`, `\t` was being re-defined somewhere.

```
\AtBeginDocument{
```

The following are restored for print when inside a `lateximage`.

For Unicode engines, only `\t` needs to be redefined:

```
\LetLtxMacro\LWR@origtie\t
```

For pdf LaTeX, additional work is required:

```
\ifPDFTeX% pdflatex or dvi latex
\LetLtxMacro\LWR@origgraveaccent\‘
\LetLtxMacro\LWR@origacuteaccent\’
\LetLtxMacro\LWR@origcircumflexaccent\^
\LetLtxMacro\LWR@origtildeaccent\~
\LetLtxMacro\LWR@origmacronaccent\=
\LetLtxMacro\LWR@origbreve\u
\LetLtxMacro\LWR@origdotaccent\.
\LetLtxMacro\LWR@origdiaeresisaccent\"
\LetLtxMacro\LWR@origdoubleacuteaccent\H
\LetLtxMacro\LWR@origcaronaccent\v
\LetLtxMacro\LWR@origdotbelowaccent\d
\LetLtxMacro\LWR@origcedillaaccent\c
\LetLtxMacro\LWR@origmacronbelowaccent\b
```

The HTML redefinitions follow.

For PDF LaTeX, Unicode diacritical marks are used:

```
\renewcommand*{\‘}[1]{#1\HTMLunicode{0300}}
\renewcommand*{\’}[1]{#1\HTMLunicode{0301}}
\renewcommand*{\^}[1]{#1\HTMLunicode{0302}}
\renewcommand*{\~}[1]{#1\HTMLunicode{0303}}
\renewcommand*{\=}[1]{#1\HTMLunicode{0304}}
\renewcommand*{\u}[1]{#1\HTMLunicode{0306}}
\renewcommand*{\.}[1]{#1\HTMLunicode{0307}}
\renewcommand*{\"}[1]{#1\HTMLunicode{0308}}
\renewcommand*{\H}[1]{#1\HTMLunicode{030B}}
\renewcommand*{\v}[1]{#1\HTMLunicode{030C}}
\renewcommand*{\d}[1]{#1\HTMLunicode{0323}}
\renewcommand*{\c}[1]{#1\HTMLunicode{0327}}
\renewcommand*{\b}[1]{#1\HTMLunicode{0331}}
\fi
```

For all engines, a Unicode diacritical tie is used:

```
\def\LWR@t#1#2{#1\HTMLunicode{0361}#2}
\renewcommand*{\t}[1]{\LWR@t#1}
```

`\LWR@restoreorigaccents` Called from `\restoreoriginalformatting` when a `latexmage` is begun.

```
\ifPDFTeX% pdflatex or dvi latex
\newcommand*{\LWR@restoreorigaccents}{%
    \LetLtxMacro\‘\LWR@origgraveaccent%
    \LetLtxMacro\’\LWR@origacuteaccent%
    \LetLtxMacro\^\LWR@origcircumflexaccent%
    \LetLtxMacro\~\LWR@origtildeaccent%
    \LetLtxMacro\=\LWR@origmacronaccent%
    \LetLtxMacro\u\LWR@origbreve%
    \LetLtxMacro\.\LWR@origdotaccent%
    \LetLtxMacro\"\LWR@origdiaeresisaccent%
    \LetLtxMacro\H\LWR@origdoubleacuteaccent%
    \LetLtxMacro\v\LWR@origcaronaccent%
    \LetLtxMacro\t\LWR@origtie%
    \LetLtxMacro\d\LWR@origdotbelowaccent%
    \LetLtxMacro\c\LWR@origcedillaaccent%
    \LetLtxMacro\b\LWR@origmacronbelowaccent%
}%
\else% XeLaTeX, LuaLaTeX:
\newcommand*{\LWR@restoreorigaccents}{%
    \LetLtxMacro\t\LWR@origtie%
}%
\fi%
}% AtBeginDocument

\end{warpHTML}
```

40 Configuration files

40.1 Decide whether to generate configuration files

Configuration files are only written if processing the print version of the document, and not processing a pstool image. pstool uses an additional compile for each image using the original document's preamble, which includes lwarp, so the lwarp configuration files are turned off if `-pstool` is part of the `\jobname`.

Default to no configuration files:

```
\LWR@excludecomment{LWRwriteconf}{writeconf}
```

Generate configuration files if print mode and not `-pstool`:

for PRINT output:

```
\begin{warpprint}
\fullexpandarg%
\IfSubStr*{\jobname}{-pstool}
    {
        \PackageInfo{lwarp}{%
            Jobname with -pstool is found.\MessageBreak
            Not generating lwarp configuration files,%
        }
    }
    {
        \PackageInfo{lwarp}{Generating lwarp configuration files,}%
        \LWR@includecomment{LWRwriteconf}{writeconf}
    }
\end{warpprint}
```

40.2 <project>_html.tex

`*_html.tex` (*file*) Used to allow an HTML version of the document to exist alongside the print version.

Config file:

```
\begin{LWRwriteconf}
\immediate\openout\LWR@quickfile=\jobname_html.tex
\immediate\write\LWR@quickfile{%
\detokenize{\PassOptionsToPackage}%
{warpHTML,BaseJobname=\jobname}{lwarp}%
}
\immediate\write\LWR@quickfile{%
\detokenize{\input}\string{\jobname.tex\string }%
}
\immediate\closeout\LWR@quickfile
\end{LWRwriteconf}
```

40.3 *lwarpmk* configuration files

Config file:

```
\begin{LWRwriteconf}
```

\LWR@lwarpconfversion The version number of the configuration file, allowing *lwarpmk* to detect an obsolete configuration file format. Incremented by one each time the configuration file format changes. (This is NOT the same as the lwarp version number.)

```
\newcommand*{\LWR@lwarpconfversion}{2}% also in lwarpmk.lua
```

40.3.1 Helper macros

\LWR@shellescapecmd The LaTeX compile option for shell escape, if used.

```
\ifshellescape
    \def\LWR@shellescapecmd{--shell-escape }
\else
    \def\LWR@shellescapecmd{}
\fi
```

\LWR@compilecmd {⟨*engine*⟩} {⟨*suffix*⟩}

Used to form the basic compilation command for a document, adding the optional shell escape.

Engine is *pdflatex*, etc. Suffix is empty or `_html`

```
\newcommand*{\LWR@compilecmd}[2]{%
    #1 \LWR@shellescapecmd \jobname#2%
}
```

\LWR@addcompilecmd {⟨*cmd*⟩} {⟨*suffix*⟩}

Adds to the compilation command.

Cmd is *dvipdfmx*, etc. Suffix is empty or `_html`

```
\newcommand*{\LWR@addcompilecmd}[2]{%
    \LWRopseq
    #1 \jobname#2%
}
```

\LWR@unknownengine Error message if not sure which LaTeX engine is being used.

```
\newcommand*{\LWR@unknownengine}{%
    \PackageError{lwarp}%
        {Unknown LaTeX engine}%
        {%
            Lwarp only knows about pdflatex, DVI latex,
            xelatex, lualatex, and upLateX.%
        }%
}
```

\LWR@latexmkvar {⟨*varname*⟩} {⟨*value*⟩}

Adds a *latexmk* variable assignment.

```
\newcommand*{\LWR@latexmkvar}[2]{%
    -e
```

```
2109     \LWRopquote%
2110     \LWRdollar #1=q/#2/%
2111     \LWRopquote
2112 }
```

`\LWR@latexmkcmd` {⟨*latexmk options*⟩}

Sets a call to *latexmk* with the given options, possibly adding `--shell-escape`, and also adding the indexing program.

```
2113 \newcommand*{\LWR@latexmkcmd}[1]{%
2114     latexmk  \space \LWR@shellescapecmd \space  #1 \space
2115     -recorder \space
2116     \LWR@latexmkvar{makeindex}{\LWR@LatexmkIndexCmd}%
2117 }
```

`\LWR@latexmkdvipdfm` {⟨*dvipdfm or dvipdfmx*⟩}

Adds the options settings for *dvipdfm* or *dvipdfmx*.

```
2118 \newcommand*{\LWR@latexmkdvipdfm}[1]{%
2119     -pdfdvi \space
2120     \LWR@latexmkvar{dvipdf}{%
2121         #1
2122         \@percentchar O
2123         -o \@percentchar D
2124         \@percentchar S%
2125     }
2126 }
```

`\LWR@compileuplatex` Sets compile options for upLaTeX with ujarticle or related classes.

```
2127 \newcommand*{\LWR@compileuplatex}{
2128     \def\LWR@tempprintlatexcmd{%
2129         \LWR@compilecmd{uplatex}{}
2130         \LWR@addcompilecmd{dvipdfmx}{}
2131     }
2132     \def\LWR@tempHTMLlatexcmd{%
2133         \LWR@compilecmd{uplatex}{_html}
2134         \LWR@addcompilecmd{dvipdfmx}{_html}
2135     }
2136 }
```

`\LWR@PrintLatexCmd` `\LWR@HTMLLatexCmd` If not set by the user, the following sets the command to use to compile the source to PDF form.

If using *latexmk*, a complicated string is created, eventually resulting in something such as:

For *xelatex* with `--shell-escape`:

```
[[latexmk   -xelatex   --shell-escape  -recorder
    -e '$makeindex = q/makeindex -s lwarp.ist/'  <jobname>_html]]
```

For *dvipdfmx*:

```
[[latexmk   -pdfdvi  -e '$dvipdf=q/dvipdfmx %O -o %D %S/'
    -recorder
    -e '$makeindex=q/makeindex -s lwarp.ist/'   <jobname>_html]]
```

For the following, temporary values are computed, but the permanent values are only set if the originals were not assigned by the user.

```
\ifbool{LWR@latexmk}{
```

For *latexmk* with *pdflatex* or *lualatex*:

```
    \ifpdf
```

For *latexmk* with *pdflatex*:

```
        \ifPDFTeX
            \def\LWR@latexcmd{\LWR@latexmkcmd{-pdf -dvi- -ps-}}
        \else
```

For *latexmk* with *lualatex*:

```
            \ifLuaTeX
                \def\LWR@latexcmd{\LWR@latexmkcmd{-lualatex}}
            \else
                \LWR@unknownengine
            \fi
        \fi
    \else% \ifpdf
```

For *latexmk* with *xelatex* or DVI *latex*:

```
        \ifXeTeX
```

For *latexmk* with *xelatex*:

```
            \def\LWR@latexcmd{\LWR@latexmkcmd{-xelatex}}
        \else% \ifXeTeX
```

For *latexmk* with DVI *latex*:

```
            \ifbool{LWR@dvipdfm}{
                \def\LWR@latexcmd{%
                    \LWR@latexmkcmd{%
                        \LWR@latexmkdvipdfm{dvipdfm}%
                    }
                }
            }{
                \ifbool{LWR@dvipdfmx}{
                    \def\LWR@latexcmd{%
                        \LWR@latexmkcmd{%
                            \LWR@latexmkdvipdfm{dvipdfmx}%
                        }
                    }
                }{
                    \def\LWR@latexcmd{\LWR@latexmkcmd{-pdfps}}
                }
```

```
            }
        \fi
    \fi% \ifpdf
```

The final assignment if *latexmk*:

```
    \def\LWR@tempprintlatexcmd{\LWR@latexcmd \space \jobname}
    \def\LWR@tempHTMLlatexcmd{\LWR@latexcmd \space \jobname_html}
}% latexmk
```

Without *latexmk*, the compiling command is simply the compiler name and the optional shell escape:

```
{% not latexmk
    \ifpdf
```

For *pdflatex* or *lualatex*:

```
        \ifPDFTeX
```

For *pdflatex*:

```
            \def\LWR@tempprintlatexcmd{\LWR@compilecmd{pdflatex}{}}
            \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{pdflatex}{_html}}
        \else
            \ifLuaTeX
```

For *lualatex*:

```
                \def\LWR@tempprintlatexcmd{\LWR@compilecmd{lualatex}{}}
                \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{lualatex}{_html}}
            \else
                \LWR@unknownengine
            \fi
        \fi
    \else% \ifpdf
```

For DVI *latex* or *xelatex*:

```
        \ifXeTeX
```

For *xelatex*:

```
            \def\LWR@tempprintlatexcmd{\LWR@compilecmd{xelatex}{}}
            \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{xelatex}{_html}}
        \else
```

For DVI *latex*. Default to *dvips*, unless told to use *dvipdfm* or *dvipdfmx*:

```
            \ifbool{LWR@dvipdfm}{
```

For DVI *latex* with *dvipdfm*:

```
                \def\LWR@tempprintlatexcmd{%
                    \LWR@compilecmd{latex}{}
                    \LWR@addcompilecmd{dvipdfm}{}
                }
                \def\LWR@tempHTMLlatexcmd{%
```

```
                    \LWR@compilecmd{latex}{_html}
                    \LWR@addcompilecmd{dvipdfm}{_html}
                }
            }{
                \ifbool{LWR@dvipdfmx}{
```

For DVI *latex* with *dvipdfmx*:

```
                    \def\LWR@tempprintlatexcmd{%
                        \LWR@compilecmd{latex}{}
                        \LWR@addcompilecmd{dvipdfmx}{}
                    }
                    \def\LWR@tempHTMLlatexcmd{%
                        \LWR@compilecmd{latex}{_html}
                        \LWR@addcompilecmd{dvipdfmx}{_html}
                    }
                }{% dvips
```

For DVI *latex* with *dvips* and *ps2pdf*:

```
                    \def\LWR@tempprintlatexcmd{%
                        \LWR@compilecmd{latex}{}
                        \LWR@addcompilecmd{dvips}{}
                       \LWR@addcompilecmd{ps2pdf -dALLOWPSTRANSPARENCY}{}.ps
                    }
                    \def\LWR@tempHTMLlatexcmd{%
                        \LWR@compilecmd{latex}{_html}
                        \LWR@addcompilecmd{dvips}{_html}
                  \LWR@addcompilecmd{ps2pdf -dALLOWPSTRANSPARENCY}{_html}.ps
                    }
                }
            }
        \fi% \ifXeTeX
    \fi% \ifpdf
}% latexmk
```

For ujarticle, utarticle, and related, using upLaTeX and *dvipdfmx*:

```
\IfClassLoadedTF{ujarticle}{\LWR@compileuplatex}{}
\IfClassLoadedTF{ujbook}{\LWR@compileuplatex}{}
\IfClassLoadedTF{ujreport}{\LWR@compileuplatex}{}
\IfClassLoadedTF{utarticle}{\LWR@compileuplatex}{}
\IfClassLoadedTF{utbook}{\LWR@compileuplatex}{}
\IfClassLoadedTF{utreport}{\LWR@compileuplatex}{}
```

Only make the setting permanent if the original was empty:

```
\ifdefempty{\LWR@PrintLatexCmd}{
    \def\LWR@PrintLatexCmd{\LWR@tempprintlatexcmd}
}{}
\ifdefempty{\LWR@HTMLLatexCmd}{
    \def\LWR@HTMLLatexCmd{\LWR@tempHTMLlatexcmd}
}{}
```

`\LWR@writeconf` {⟨*filename*⟩}

Common code for each of `lwarpmk.conf` and `<project>.lwarpmkconf`. Each entry is a variable name, the equal sign, and a quoted string inside `[[` and `]]`, which are *lua*'s long quote characters, allowing the use of single and double quotes inside.

```
\newcommand{\LWR@writeconf}[1]{
\ifcsdef{LWR@quickfile}{}{\newwrite{\LWR@quickfile}}
\immediate\openout\LWR@quickfile=#1
\immediate\write\LWR@quickfile{confversion = [[\LWR@lwarpconfversion]]}
\ifbool{usingOSWindows}{
    \immediate\write\LWR@quickfile{opsystem = [[Windows]]}
}{
    \immediate\write\LWR@quickfile{opsystem = [[Unix]]}
}
\immediate\write\LWR@quickfile{sourcename = [[\jobname]]}
\immediate\write\LWR@quickfile{homehtmlfilename = [[\HomeHTMLFilename]]}
\immediate\write\LWR@quickfile{htmlfilename = [[\HTMLFilename]]}
\immediate\write\LWR@quickfile{imagesdirectory = [[\LWR@ImagesDirectory]]}
\immediate\write\LWR@quickfile{imagesname = [[\LWR@ImagesName]]}
\immediate\write\LWR@quickfile{latexmk = [[\ifbool{LWR@latexmk}{true}{false}]]}
\immediate\write\LWR@quickfile{printlatexcmd = [[\LWR@PrintLatexCmd]]}
\immediate\write\LWR@quickfile{HTMLlatexcmd = [[\LWR@HTMLLatexCmd]]}
\immediate\write\LWR@quickfile{printindexcmd = [[\LWR@PrintIndexCmd]]}
\immediate\write\LWR@quickfile{HTMLindexcmd = [[\LWR@HTMLIndexCmd]]}
\immediate\write\LWR@quickfile{latexmkindexcmd = [[\LWR@LatexmkIndexCmd]]}
\immediate\write\LWR@quickfile{glossarycmd = [[\LWR@GlossaryCmd]]}
\immediate\write\LWR@quickfile{pdftotextenc = [[\LWR@pdftotextEnc]]}
\immediate\closeout\LWR@quickfile
}

\end{LWRwriteconf}
```

40.3.2 `lwarpmk.conf`

`lwarpmk.conf` (*file*)

`lwarpmk.conf` is automatically (re-)created by the lwarp package when executing
`pdflatex <project.tex>`,
or similar for *xelatex* or *lualatex*, in print-document generation mode, which is the default unless the `warpHTML` option is given. `lwarpmk.conf` is then used by the utility *lwarpmk*.

Config file:

```
\begin{LWRwriteconf}

\AtBeginDocument{\LWR@writeconf{lwarpmk.conf}}

\end{LWRwriteconf}
```

40.3.3 `<project>.lwarpmkconf`

`project.lwarpmkconf` (*file*)

A project-specific configuration file for *lwarpmk*.

The `makeindex` and `xindy` options have already been handled for `lwarp.conf`.

Config file:

```
\begin{LWRwriteconf}

\AtBeginDocument{\LWR@writeconf{\jobname.lwarpmkconf}}

\end{LWRwriteconf}
```

40.4 lwarp.css

lwarp.css (*file*) This is the base CSS layer used by lwarp.

This must be present both when compiling the project and also when distributing the HTML files.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{lwarp.css}
/*
  CSS stylesheet for the LaTeX Lwarp package
  Copyright 2016-2022 Brian Dunn — BD Tech Concepts LLC
*/


/* a fix for older browsers: */
header, section, footer, aside, nav, main,
    article, figure { display: block; }


A:link {color:#000080 ; text-decoration: none ; }
A:visited {color:#800000 ; }
A:hover {color:#000080 ; text-decoration: underline ;}
A:active {color:#800000 ; }

a.tocbook {display: inline-block ; margin-left: 0em ;
    font-weight: bold ; margin-top: 1ex ; margin-bottom: 1ex ; }
a.tocpart {display: inline-block ; margin-left: 0em ;
    font-weight: bold ;}
a.tocchapter {display: inline-block ; margin-left: 0em ;
    font-weight: bold ;}
a.tocsection {display: inline-block ; margin-left: 1em ;
    text-indent: -.5em ; font-weight: bold ; }
a.tocsubsection {display: inline-block ; margin-left: 2em ;
    text-indent: -.5em ; }
a.tocsubsubsection {display: inline-block ; margin-left: 3em ;
    text-indent: -.5em ; }
a.tocparagraph {display: inline-block ; margin-left: 4em ;
    text-indent: -.5em ; }
a.tocsubparagraph {display: inline-block ; margin-left: 5em ;
    text-indent: -.5em ; }
a.tocfigure {margin-left: 0em}
a.tocsubfigure {margin-left: 2em}
a.toctable {margin-left: 0em}
a.tocsubtable {margin-left: 2em}
a.toctheorem {margin-left: 0em}
a.toclstlisting {margin-left: 0em}

body {
    font-family: "DejaVu Serif", "Bitstream Vera Serif",
        "Lucida Bright", Georgia, serif;
    background: #FAF7F4 ;
    color: black ;
    margin:0em ;
    padding:0em ;
    font-size: 100% ;
    line-height: 1.2 ;
}

p {margin: 1.5ex 0em 1.5ex 0em ;}
```

```
table p {margin: .5ex 0em .5ex 0em ;}

/* Holds a section number */
span.sectionnumber { margin-right: 0em }

/* Inserted in front of index lines */
span.indexitem {margin-left: 0em}
span.indexsubitem {margin-left: 2em}
span.indexsubsubitem {margin-left: 4em}
div.indexheading {margin-top: 2ex ; font-weight: bold}

div.hidden, span.hidden { display: none ; }

kbd, span.texttt, p span.texttt {
    font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
    font-size: 100% ;
}

pre { padding: 3pt ; }

span.strong, span.textbf, div.strong, div.textbf, table td.tdbfseries { font-weight: bold; }

span.textit, div.textit, table td.tditshape { font-style: italic; }

table td.tdbfit { font-weight: bold ; font-style:italic }

span.textnormal, div.textnormal {
    font-weight: normal;
    font-style: normal;
    font-variant: normal;
    font-variant-numeric: normal ;
    font-family: "DejaVu Serif", "Bitstream Vera Serif",
        "Lucida Bright", Georgia, serif;
}

span.textmd, div.textmd { font-weight: normal; }

span.textup, div.textup {
    font-style: normal;
    font-variant: normal;
    font-variant-numeric: normal ;
}

span.textsc, div.textsc {
    font-variant: small-caps;
    font-variant-numeric: oldstyle-nums ;
}

span.textulc, div.textulc {
    font-variant: normal ;
    font-variant-numeric: normal ;
}

span.textsl, div.textsl { font-style: oblique; }

span.textrm, div.textrm {
    font-family: "DejaVu Serif", "Bitstream Vera Serif",
    "Lucida Bright", Georgia, serif;
```

```
}

span.textsf, div.textsf {
     font-family: "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
}

/* nfssext-cfr lining figures */
span.textln, div.textln {
    font-variant-numeric: lining-nums ;
}

/* nfssext-cfr proportional figures */
span.textp, div.textp {
    font-variant-numeric: proportional-nums ;
}

/* nfssext-cfr tabular figures */
span.textt, div.textt {
    font-variant-numeric: tabular-nums ;
}

/* nfssext-cfr font weights */
span.textdb, div.textdb {
    font-weight: 500 ;
}

span.textsb, div.textsb {
    font-weight: 600 ;
}

span.texteb, div.texteb {
    font-weight: 800 ;
}

span.textub, div.textub {
    font-weight: 900 ;
}

span.textlg, div.textlg {
    font-weight: 300 ;
}

span.textel, div.textel {
    font-weight: 200 ;
}

span.textul, div.textul {
    font-weight: 100 ;
}



span.textcircled { border: 1px solid black ; border-radius: 1ex ; }

span.underline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
}

```

```
span.overline {
    text-decoration: overline ;
    text-decoration-skip: auto ;
}

div.hrule { border-top: 1px solid silver }


/* for vertical text: */
div.verticalrl { writing-mode: vertical-rl }
div.horizontaltb { writing-mode: horizontal-tb }


/* for diagbox */
div.diagboxtitleN { border-bottom: 1px solid gray }
div.diagboxtitleS { border-top: 1px solid gray }

div.diagboxE {
    padding-left: 2em ;
    text-align: right ;
}

div.diagboxW {
    padding-right: 2em ;
    text-align: left ;
}



/* For realscripts */
.supsubscript {
    display: inline-block;
    text-align:left ;
}

.supsubscript sup,
.supsubscript sub {
    position: relative;
    display: block;
    font-size: .7em;
    line-height: 1;
}

.supsubscript sup {
    top: .3em;
}

.supsubscript sub {
    top: .3em;
}

div.attribution p {
    text-align: right ;
    font-size: 80%
}

span.poemtitle {
  font-size: 120% ; font-weight: bold;
}

```

```
pre.tabbing {
    font-family: "Linux Libertine Mono O", "Lucida Console",
        "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
        "Liberation Mono", "FreeMono", "Andale Mono",
        "Nimbus Mono L", "Courier New", monospace;
}

blockquote {
    display: block ;
    margin-left: 2em ;
    margin-right: 2em ;
}

/* quotchap is for the quotchap package */
div.quotchap {
    display: block ;
    font-style: oblique ;
    overflow-x: auto ;
    margin-left: 2em ;
    margin-right: 2em ;
}

blockquote p, div.quotchap p {
    line-height: 1.5;
    text-align: left ;
    font-size: .85em ;
}

/* qauthor is for the quotchap package */
div.qauthor {
  display: block ;
  text-align: right ;
  margin-left: auto ;
  margin-right: 2em ;
  font-size: 80% ;
  font-variant: small-caps;
}

div.qauthor p {
  text-align: right ;
}

div.epigraph, div.dictum {
  line-height: 1.2;
    text-align: left ;
    padding: 3ex 1em 0ex 1em ;
/*     margin: 3ex auto 3ex auto ; */ /* Epigraph centered */
    margin: 3ex 1em 3ex auto ; /* Epigraph to the right */
/*    margin: 3ex 1em 3ex 1em ; */ /* Epigraph to the left */
    font-size: .85em ;
    max-width: 27em ;
}

div.epigraphsource, div.dictumauthor {
    text-align:right ;
    margin-left:auto ;
/*     max-width: 50% ; */
    border-top: 1px solid #A0A0A0 ;
    padding-bottom: 3ex ;
    line-height: 1.2;
```

```
}

div.epigraph p, div.dictum p { padding: .5ex ; margin: 0ex ;}
div.epigraphsource p, div.dictumauthor p { padding: .5ex 0ex 0ex 0ex ; margin: 0ex ;}
div.dictumauthor { font-style:italic }


/* copyrightbox package: */
div.copyrightbox { margin: .5ex .5em }
div.copyrightbox p {margin: 0px .5em ; padding: 0px}
div.copyrightboxnote {text-align: left ; font-size: 60%}


/* lettrine package: */
span.lettrine { font-size: 4ex ; float: left ; }
span.lettrinetext { font-variant: small-caps ; }

/* ulem, soul, umoline packages: */
span.uline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
}

span.uuline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
    text-decoration-style: double ;
}

span.uwave {
    text-decoration: underline ;
    text-decoration-skip: auto ;
    text-decoration-style: wavy ;
}

span.sout {
    text-decoration: line-through ;
}

span.oline {
    text-decoration: overline ;
    text-decoration-skip: auto ;
}

span.xout {
    text-decoration: line-through ;
}

span.dashuline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
    text-decoration-style: dashed ;
}

span.dotuline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
    text-decoration-style: dotted ;
}

```

```
span.letterspacing { letter-spacing: .2ex ; }

span.capsspacing {
    font-variant: small-caps ;
    letter-spacing: .1ex ;
}

span.highlight { background: #F8E800 ; }


/* keystroke package: */
span.keystroke {
    border-style: outset ;
    padding: 0pt .5em 0pt .5em ;
}


html body {
  margin: 0 ;
  line-height: 1.2;
}


body div {
  margin: 0ex;
}


div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
{
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    font-style: normal ;
    font-weight: bold ;
    text-align: left ;
}

h1 {    /* title of the entire website, used on each page */
    text-align: center ;
    font-size: 2.5em ;
    padding: .4ex 0em 0ex 0em ;
}

div.book {
    text-align: center ;
    font-size: 2.325em ;
    padding: .4ex 0em 0ex 0em ;
}

h2 { font-size: 2.25em }
h3 { font-size: 2em }
h4 { font-size: 1.75em }
h5 { font-size: 1.5em }
h6 { font-size: 1.25em }
span.paragraph {font-size: 1em ; font-variant: normal ;
    margin-right: 1em ; }
span.subparagraph {font-size: 1em ; font-variant: normal ;
    margin-right: 1em ;  }
```

```
2688
2689 div.minisec {
2690     font-family: "DejaVu Sans", "Bitstream Vera Sans",
2691         Geneva, Verdana, sans-serif ;
2692     font-style: normal ;
2693     font-weight: bold ;
2694     text-align: left ;
2695 }
2696
2697 h1 {
2698   margin: 0ex 0em 0ex 0em ;
2699   line-height: 1.3;
2700   text-align: center ;
2701 }
2702
2703 h2 {
2704   margin: 1ex 0em 1ex 0em ;
2705   line-height: 1.3;
2706   text-align: center ;
2707 }
2708
2709 h3 {
2710   margin: 3ex 0em 1ex 0em ;
2711   line-height: 1.3;
2712 }
2713
2714 h4 {
2715   margin: 3ex 0em 1ex 0em ;
2716   line-height: 1.3;
2717 }
2718
2719 h5 {
2720   margin: 3ex 0em 1ex 0em ;
2721   line-height: 1.3;
2722 }
2723
2724 h6 {
2725   margin: 3ex 0em 1ex 0em ;
2726   line-height: 1.3;
2727 }
2728
2729
2730 div.titlepage {
2731   text-align: center ;
2732 }
2733
2734 .footnotes {
2735     text-align: left ;
2736     font-size: .85em ;
2737     margin: 3ex 2em 0ex 2em ;
2738     border-top: 1px solid silver ;
2739 }
2740
2741 .marginpar, .marginparblock {
2742     max-width: 50%;
2743     float: right ;
2744     clear: both ;
2745     text-align: left ;
2746     margin: 1ex 0.5em 1ex 1em ;
2747     padding: 1ex 0.5em 1ex 0.5em ;
```

```
2748     font-size: 85% ;
2749     border-top: 1px solid silver ;
2750     border-bottom: 1px solid silver ;
2751     overflow-x: auto ;
2752 }
2753
2754 .marginpar br { margin-bottom: 2ex ; }
2755
2756 div.marginblock, div.marginparblock {
2757     max-width:50%;
2758     min-width: 10em; /* room for caption */
2759     float:right;
2760     text-align:left;
2761     margin: 1ex 0.5em 1ex 1em ;
2762     padding: 1ex 0.5em 1ex 0.5em ;
2763     overflow-x: auto;
2764 }
2765
2766 div.marginblock div.minipage,
2767 div.marginparblock div.minipage {
2768     display: inline-block ;
2769     margin: 0pt auto 0pt auto ;
2770 }
2771
2772 div.marginblock div.minipage p ,
2773 div.marginparblock div.minipage p
2774     { font-size: 85%}
2775
2776 div.marginblock br ,
2777 div.marginparblock br
2778     { margin-bottom: 2ex ; }
2779
2780 main.bodycontainer {
2781     float: left ;
2782     width: 80% ;
2783 }
2784
2785 div.bodywithoutsidetoc main.bodycontainer {
2786     float: none ;
2787     width: 100% ;
2788 }
2789
2790 section.textbody div.footnotes{
2791     margin: 1ex 2em 2ex 2em ;
2792     border-bottom: 2px solid silver ;
2793 }
2794
2795 .footnoteheader {
2796     border-top: 2px solid silver ;
2797     margin-top: 3ex ;
2798     padding-top: 1ex ;
2799     font-weight: bold ;
2800 }
2801
2802 .mpfootnotes {
2803     text-align: left ;
2804     font-size: .85em ;
2805     margin-left: 1em ;
2806     border-top: 1px solid silver ;
2807 }
```

```
2808
2809 /* Remove footnote top border in the title page. */
2810 div.titlepage div.mpfootnotes {
2811     border-top: none ;
2812 }
2813
2814
2815
2816 ul, ol {
2817   margin: 1ex 1em 1ex 0em;
2818   line-height: 1.2;
2819 }
2820
2821 body dir, body menu {
2822   margin: 3ex 1em 3ex 0em;
2823   line-height: 1.2;
2824 }
2825
2826 li { margin: 0ex 0em 1ex 0em; }
2827
2828 li.p { display: inline ; }
2829
2830 html {
2831   margin: 0;
2832   padding: 0;
2833 }
2834
2835 .programlisting {
2836   font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2837         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2838         "Courier New", monospace;
2839   margin: 1ex 0ex 1ex 0ex ;
2840   padding: .5ex 0pt .5ex 0pt ;
2841   overflow-x: auto;
2842 }
2843
2844 section.textbody>pre.programlisting {
2845 border-top: 1px solid silver ;
2846 border-bottom: 1px solid silver ;
2847 }
2848
2849
2850 div.displaymath {
2851     text-align: center ;
2852 }
2853
2854 div.displaymathnumbered {
2855     text-align: right ;
2856     margin-left: 5% ;
2857     margin-right: 5% ;
2858     min-width: 2.5in ;
2859 }
2860
2861 @media all and (min-width: 400px) {
2862     div.displaymathnumbered {
2863         margin-left: 10% ;
2864         margin-right: 10% ;
2865     }
2866 }
2867
```

```
@media all and (min-width: 800px) {
    div.displaymathnumbered {
        margin-right: 20% ;
    }
}

@media all and (min-width: 1200px) {
    div.displaymathnumbered {
        margin-right: 30% ;
    }
}


.inlineprogramlisting {
  font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
  overflow-x: auto;
}

span.listinglabel {
    display: inline-block ;
    font-size: 70% ;
    width: 4em ;
    text-align: right ;
    margin-right: 2em ;
}

div.abstract {
  margin: 2em 5% 2em 5% ;
  padding: 1ex 1em 1ex 1em ;
/*  font-weight: bold ; */
  font-size: 90% ;
    text-align: left ;
}

div.abstract dl {line-height:1.5;}
div.abstract dt {color:#304070;}

div.abstracttitle{
    font-family: "URW Classico", Optima, "Linux Biolinum O",
        "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    font-weight:bold;
    font-size:1.25em;
    text-align: center ;
}

span.abstractrunintitle{
    font-family: "URW Classico", Optima, "Linux Biolinum O",
        "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    font-weight:bold;
}


.verbatim {
    overflow-x: auto ;
}

```

```
.alltt {
    overflow-x: auto ;
}


.bverbatim {
    margin: 1ex 0pt 1ex 0pt ;
    padding: .5ex 0pt .5ex 0pt ;
    overflow-x: auto ;
}

.lverbatim {
    margin: 1ex 0pt 1ex 0pt ;
    padding: .5ex 0pt .5ex 0pt ;
    overflow-x: auto ;
}

.fancyvrb {
    font-size:.85em ;
    margin: 3ex 0pt 3ex 0pt
}

.fancyvrblabel {
    font-size: .85em ;
    text-align: center ;
    font-weight: bold ;
    margin-top: 1ex ;
    margin-bottom: 1ex ;
}


.verse {
    font-family: "Linux Libertine Mono O", "Lucida Console",
        "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
        "Liberation Mono", "FreeMono", "Andale Mono",
        "Nimbus Mono L", "Courier New", monospace;
    margin-left: 1em ;
}


div.singlespace { line-height: 1.2 ; }
div.onehalfspace { line-height: 1.5 ; }
div.doublespace { line-height: 2 ; }


/* Word processor format output: */
div.wpfigure { border: 1px solid red ; margin: .5ex ; padding: .5ex ; }
div.wptable { border: 1px solid blue ; margin: .5ex ; padding: .5ex ; }
div.wpminipage { border: 1px solid green ; margin: .5ex ; padding: .5ex ;}



/* Minipage environments, vertically aligned to top, center, bottom: */
.minipage, .fminipage, .fcolorminipage {
    /* display: inline-block ; */
        /* Mini pages which follow each other will be tiled. */
    text-align:left;
    margin: .25em .25em .25em .25em;
    padding: .25em .25em .25em .25em;
```

```
2988     display: inline-flex;
2989     flex-direction: column ;
2990     overflow: auto;
2991 }
2992
2993 .inlineminipage {
2994     display: inline-block ;
2995     text-align: left
2996 }
2997
2998 /* Paragraphs in the flexbox did not collapse their margins. */
2999 /* Have not yet researched this. */
3000 .minipage p {margin: .75ex 0em .75ex 0em ;}
3001
3002 .fboxBlock .minipage, .colorbox .minipage, .colorboxBlock .minipage,
3003 .fcolorbox .minipage, .fcolorboxBlock .minipage
3004     {border: none ; background: none;}
3005
3006 .fbox, .fboxBlock { border: 1px solid black ; padding: 4pt }
3007
3008 .fbox, .fboxBlock, .fcolorbox, .fcolorboxBlock, .colorbox, .colorboxBlock,
3009 .fminipage, .fcolorminipage
3010     {display: inline-block}
3011
3012 .shadowbox, .shabox {
3013    border: 1px solid black;
3014     box-shadow: 3px 3px 3px #808080 ;
3015      border-radius: 0px ;
3016     padding: .4ex .3em .4ex .3em ;
3017     margin: 0pt .3ex 0pt .3ex ;
3018   display: inline-block ;
3019 }
3020
3021 .doublebox {
3022    border: 3px double black;
3023      border-radius: 0px ;
3024     padding: .4ex .3em .4ex .3em ;
3025     margin: 0pt .3ex 0pt .3ex ;
3026   display: inline-block ;
3027 }
3028
3029 .ovalbox, .Ovalbox {
3030    border: 1px solid black;
3031      border-radius: 1ex ;
3032     padding: .4ex .3em .4ex .3em ;
3033     margin: 0pt .3ex 0pt .3ex ;
3034   display: inline-block ;
3035 }
3036
3037 .Ovalbox { border-width: 2px ; }
3038
3039 .framebox {
3040    border: 1px solid black;
3041      border-radius: 0px ;
3042     padding: .3ex .2em 0ex .2em ;
3043     margin: 0pt .1ex 0pt .1ex ;
3044   display: inline-block ;
3045 }
3046
3047
```

```
/* mdframed, tcolorbox, shadebox packages */
.mdframed, .tcolorbox, .shadebox {
    padding: 0ex ;
    margin: 2ex 0em 2ex 0em ;
    border: 1px solid black ;
}

.tcolorbox {
    border-radius: 10pt ;
    margin: 2ex 1em 2ex 1em ;
}

.mdframed p, .tcolorbox p { padding: 0ex .5em 0ex .5em ; }

.mdframed dl, .tcolorbox dl { padding: 1ex .5em 0ex .5em ; }

.mdframedtitle, .tcolorboxtitle {
    padding: .5ex 0pt 0pt 0pt ;
    border-radius: 10pt 10pt 0pt 0pt ;
    display: block ;
    margin-bottom: 1ex ;
    border-bottom: 1px solid silver ;
}

.tcolorboxsubtitle .tcolorbox {
    margin: 2ex 0em 2ex 0em ;
    border-radius: 0pt ;
}

.mdframedsubtitle {
    display: block ;
}

.mdframedsubsubtitle {
    display: block ;
}

.mdtheorem {
    padding: 0ex .5em 0ex .5em ;
    margin: 3ex 5% 3ex 5% ;
}


/* framed package */
.framed, pre.boxedverbatim, fcolorbox {
    margin: 3ex 0em 3ex 0em ;
   border: 1px solid black;
     border-radius: 0px ;
    padding: .3ex 1em 0ex 1em ;
  display: block ;
}

.shaded {
    margin: 3ex 0em 3ex 0em ;
    padding: .3ex 1em .3ex 1em ;
    display: block ;
}

.snugframed {
    margin: 3ex 0em 3ex 0em ;
```

```
3108    border: 1px solid black;
3109      border-radius: 0px ;
3110   display: block ;
3111 }
3112
3113 .framedleftbar {
3114     margin: 3ex 0em 3ex 0em ;
3115    border-left: 3pt solid black;
3116      border-radius: 0px ;
3117     padding: .3ex .2em .3ex 1em ;
3118   display: block ;
3119 }
3120
3121 .framedtitle {
3122     margin: 0em ;
3123     padding: 0em ;
3124     font-size: 130%
3125 }
3126
3127 .framedtitle p { padding: .3em }
3128
3129
3130 /* For the niceframe package: */
3131
3132 div.niceframe, div.curlyframe, div.artdecoframe, div.generalframe {
3133     padding: 1ex ;
3134     margin: 2ex auto ;
3135     border-radius: 2ex ;
3136 }
3137
3138 div.niceframe {
3139     border: 6px groove black ;
3140 }
3141
3142 div.curlyframe {
3143     border-left: 3px dotted black ;
3144     border-right: 3px dotted black ;
3145     border-radius: 6ex ;
3146 }
3147
3148 div.artdecoframe {
3149     border-left: 10px double black ;
3150     border-right: 10px double black ;
3151     border-radius: 6ex ;
3152 }
3153
3154 div.generalframe {
3155     border: 6px groove black ;
3156 }
3157
3158
3159 /* For beamerarticle: */
3160 div.beamerframe {
3161     margin: 3ex 1em 3ex 1em ;
3162    border: 1px solid gray;
3163      border-radius: 0px ;
3164     padding: .3ex 1em 0ex 1em ;
3165   display: block ;
3166 }
3167
```

```
3168
3169 dl {
3170   margin: 1ex 2em 1ex 0em;
3171   line-height: 1.3;
3172 }
3173
3174 dl dt {
3175     display: block ;
3176     float:left ;
3177     font-weight: bold;
3178     padding-right: 1em ;
3179 }
3180
3181 dl dd { display: block ; }
3182
3183 dl dd:after { content: "" ; display: block ; clear: both }
3184
3185 dl dd p { margin-top: 0em; }
3186
3187 dd ul, dd ol, dd dl {
3188     clear: both ;
3189 /*    padding-top: 1ex ; */
3190 }
3191
3192
3193 nav {
3194     font-family: "URW Classico", Optima, "Linux Biolinum O",
3195         "DejaVu Sans", "Bitstream Vera Sans",
3196         Geneva, Verdana, sans-serif ;
3197     margin-bottom: 4ex ;
3198 }
3199
3200 nav p {
3201     line-height: 1.2 ;
3202     margin-top:.5ex ;
3203     margin-bottom:.5ex;
3204     font-size: .9em ;
3205 }
3206
3207
3208
3209 img, img.hyperimage, img.borderimage {
3210     max-width: 600px;
3211     border: 1px solid silver;
3212     box-shadow: 3px 3px 3px #808080 ;
3213     padding: .5% ;
3214     margin: .5% ;
3215     background: none ;
3216 }
3217
3218 img.inlineimage{
3219     padding: 0px ;
3220     box-shadow: none ;
3221     border: none ;
3222     background: none ;
3223     margin: 0px ;
3224     display: inline-block ;
3225     border-radius: 0px ;
3226 }
3227
```

```
3228 img.logoimage{
3229     max-width: 300px ;
3230     box-shadow: 3px 3px 3px #808080 ;
3231     border: 1px solid black ;
3232     background:none ;
3233     padding:0 ;
3234     margin:.5ex ;
3235     border-radius: 10px ;
3236 }
3237
3238
3239 .section {
3240 /*
3241     To have each section float relative to each other:
3242 */
3243 /*
3244     display: block ;
3245     float: left ;
3246     position: relative ;
3247     background: white ;
3248     border: 1px solid silver ;
3249     padding: .5em ;
3250 */
3251     margin: 0ex .5em 0ex .5em ;
3252     padding: 0 ;
3253 }
3254
3255
3256 figure {
3257     margin: 5ex auto 5ex auto ;
3258     padding: 1ex 1em 1ex 1em ;
3259     overflow-x: auto ;
3260 }
3261
3262
3263 /* To automatically center images in figures: */
3264 /*
3265 figure img.inlineimage {
3266     margin: 0ex auto 0ex auto ;
3267     display: block ;
3268 }
3269 */
3270
3271 /* To automatically center minipages in figures: */
3272 /*
3273 figure div.minipage, figure div.minipage div.minipage {
3274     margin: 1ex auto 1ex auto ;
3275     display: block ;
3276 }
3277 */
3278
3279 figure figure { margin: 0pt }
3280
3281 figure div.minipage p { font-size: 85% ; }
3282
3283 figure.subfigure, figure.subtable  {
3284     display: inline-block ; margin: 3ex 1em 3ex 1em ;
3285 }
3286
3287 div.figurecaption .minipage { margin:0 ; padding: 0 }
```

```
3288
3289 /* for subcaptions: */
3290 figure div.minipage div.figurecaption {
3291     max-width: 100% ;   /* fallback if min() does not work */
3292     max-width: min(30em,100%)
3293 }
3294
3295 div.minipage figure { border: none ; box-shadow: none ; }
3296 div.minipage figure.table { margin: 0ex }
3297 div.minipage div.footnotes { margin: 1ex 2em 0ex 2em }
3298
3299 div.floatrow { text-align: center; }
3300
3301 div.floatrow figure { display: inline-block ; margin: 1ex 2% ; }
3302
3303 div.floatfoot { font-size: .85em ;
3304     border-top: 1px solid silver ; line-height: 1.2 ; }
3305
3306 /* Center if only one line, "start" align if more than one line: */
3307 div.figurecaption , .lstlistingtitle {
3308     font-size: .85em ;
3309     font-weight: bold ;
3310     text-align: start ;
3311     margin: 1ex auto;
3312     width: max-content;
3313     max-width: 100%;
3314 }
3315
3316 /* A marginblock is small, so always center and don't mess with the width. */
3317 div.marginblock div.figurecaption {
3318     width: 100% ;
3319     text-align: center ;
3320 }
3321
3322 figure.subfigure div.figurecaption, figure.subtable div.figurecaption  {
3323     border-bottom: none ; background: none ;
3324 }
3325
3326 div.nonfloatcaption {
3327     margin: 1ex auto 1ex auto ;
3328     font-size: .85em ;
3329     text-align: center ;
3330     font-weight: bold ;
3331 }
3332
3333 /* For a \RawCaption inside a minipage inside a figure's floatrow: */
3334 figure div.floatrow div.minipage div.figurecaption {
3335     border: none ;
3336     background: none ;
3337 }
3338
3339
3340 /* For packages such as float, rotfloat, and algorithm2e: */
3341
3342 figure.boxed, figure.boxruled {
3343     border: 1px solid black ;
3344 }
3345
3346 figure.ruled {
3347     border-top: 1px solid black ;
```

```
3348     border-bottom: 1px solid black ;
3349     border-left: 0px ;
3350     border-right: 0px ;
3351     border-radius: 0px ;
3352     background: none ;
3353     box-shadow: none ;
3354 }
3355
3356 figure.ruled div.figurecaption, figure.boxruled div.figurecaption {
3357     border-top: 1px solid silver ;
3358     border-bottom: 1px solid silver ;
3359 }
3360
3361
3362 table {
3363     margin: 1ex auto 1ex auto ;
3364     border-collapse: separate ;
3365     border-spacing: 0px ;
3366     line-height: 1.3 ;
3367     }
3368
3369 table > tbody > tr.hline > td {border-top: 1px solid #808080 ; margin-top: 0ex ;
3370     margin-bottom: 0ex ; } /* for \hline */
3371
3372 tr.tbrule td {border-top: 1px solid black ; margin-top: 0ex ;
3373     margin-bottom: 0ex ; } /* for \toprule, \bottomrule */
3374
3375 td {padding: .5ex .5em .5ex .5em ;}
3376
3377 table td.tdl { text-align: left ; vertical-align: middle ; }
3378 table td.tdc { text-align: center ; vertical-align: middle ; }
3379 table td.tdat { text-align: center ; vertical-align: middle ; padding: 0px ; margin: 0px ; }
3380 table td.tdbang { text-align: center ; vertical-align: middle ; }
3381 table td.tdr { text-align: right ; vertical-align: middle ; }
3382 table td.tdp { text-align: left ; vertical-align: bottom ; }
3383 table td.tdm { text-align: left ; vertical-align: middle ; }
3384 table td.tdb { text-align: left ; vertical-align: top ; }
3385
3386 table td.tvertbarl { border-left: 1px solid black }
3387 table td.tvertbarldouble { border-left: 4px double black }
3388 table td.tvertbarr { border-right: 1px solid black }
3389 table td.tvertbarrdouble { border-right: 4px double black }
3390
3391 table td.tvertbarldash { border-left: 1px dashed black }
3392 table td.tvertbarldoubledash { border-left: 2px dashed black }
3393 table td.tvertbarrdash { border-right: 1px dashed black }
3394 table td.tvertbarrdoubledash { border-right: 2px dashed black }
3395
3396 table td.tdcenter { text-align: center}
3397 table td.tdleft { text-align: left}
3398 table td.tdright { text-align: right}
3399
3400
3401 /* for cmidrules: */
3402 table td.tdrule {
3403     border-top: 1px solid #A0A0A0 ;
3404 }
3405
3406 table td.tdrulel {
3407     border-top-left-radius:.5em ;
```

```
3408     border-top: 1px solid #A0A0A0 ;
3409 }
3410
3411 table td.tdruler {
3412     border-top-right-radius:.5em ;
3413     border-top: 1px solid #A0A0A0 ;
3414 }
3415
3416 table td.tdrulelr {
3417     border-top-left-radius:.5em ;
3418     border-top-right-radius:.5em ;
3419     border-top: 1px solid #A0A0A0 ;
3420 }
3421
3422
3423 /* Margins of paragraphs inside table cells: */
3424 td.tdp p , td.tdprule p , td.tdP p , td.tdPrule p { padding-top: 1ex ;
3425     padding-bottom: 1ex ; margin: 0ex ; }
3426 td.tdm p , td.tmbrule p , td.tdM p , td.tdMrule p { padding-top: 1ex ;
3427     padding-bottom: 1ex ; margin: 0ex ; }
3428 td.tdb p , td.tdbrule p , td.tdB p , td.tdBrule p { padding-top: 1ex ;
3429     padding-bottom: 1ex ; margin: 0ex ; }
3430
3431 td.tdp , td.tdprule , td.tdP , td.tdPrule
3432     { padding: 0ex .5em 0ex .5em ; }
3433 td.tdm , td.tdmrule , td.tdM , td.tdMrule
3434     { padding: 0ex .5em 0ex .5em ; }
3435 td.tdb , td.tdbrule , td.tdB , td.tdBrule
3436     { padding: 0ex .5em 0ex .5em ; }
3437
3438
3439 /* table notes: */
3440 .tnotes {
3441     margin: 0ex 5% 1ex 5% ;
3442     padding: 0.5ex 1em 0.5ex 1em;
3443     font-size:.80em;
3444     text-align: left ;
3445 }
3446
3447 .minipage .tnotes {
3448     margin: 0pt ;
3449     padding: 0pt ;
3450 }
3451
3452 .tnotes dl dt p {margin-bottom:0px;}
3453
3454 .tnoteitemheader {margin-right: 1em;}
3455
3456
3457 /* for colortbl and cell color */
3458 div.cellcolor {
3459     width: 100% ;
3460     padding: .5ex .5em .5ex .5em ;
3461     margin: -.5ex -.5em -.5ex -.5em ;
3462 }
3463
3464
3465 /* for lyluatex */
3466 span.lyluatex {
3467     display: inline-block ;
```

```
}

div.lyluatex p span.lateximagesource img {
    display: block ;
    margin-top: 3ex ;
    margin-bottom: 3ex ;
}


/* for bigdelim */
.ldelim, .rdelim { font-size: 200% }


/* center, flushleft, flushright environments */
div.center{text-align:center;}
div.center table {margin-left:auto;margin-right:auto;}
div.flushleft{text-align:left;}
div.flushleft table {margin-left:0em ; margin-right:auto;}
div.flushright{text-align:right;}
div.flushright table {margin-left:auto ; margin-right: 0em ;}


/* Fancybox */
div.Btrivlist table tr td {
    padding: .2ex 0em ;
}


/* program listing callouts: */
span.callout {
     font-family: "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
    border-radius: .5em;
    background-color:black;
    color:white;
    padding:0px .25em 0px .25em;
    margin: 0 ;
    font-weight: bold;
    font-size:.72em ;
}

div.programlisting pre.verbatim span.callout{
    font-size: .85em ;
}

span.verbatim {
    font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
}



div.titlehead
{
    text-align: left ;
    font-style: normal ;
    font-weight: normal ;
    font-style: normal ;
    font-size: .8em ;
```

```
3528     margin: 1ex 0em 1ex 0em ;
3529 }
3530
3531 div.subject
3532 {
3533     text-align: center ;
3534     font-style: normal ;
3535     font-weight: bold ;
3536     font-style: normal ;
3537     font-size: .8em ;
3538     margin: 1ex 0em 1ex 0em ;
3539 }
3540
3541 div.published
3542 {
3543     text-align: center ;
3544     font-variant: normal ;
3545     font-style: italic ;
3546     font-size: 1em ;
3547     margin: 1ex 0em 1ex 0em ;
3548 }
3549
3550 div.subtitle
3551 {
3552     text-align: center ;
3553     font-variant: normal ;
3554     font-style: italic ;
3555     font-size: 1.25em ;
3556     margin: 1ex 0em 1ex 0em ;
3557 }
3558
3559 div.subtitle p { margin: 1ex ; }
3560
3561 div.author
3562 {
3563     text-align: center ;
3564     font-variant: normal ;
3565     font-style: normal ;
3566     font-size: 1em ;
3567     margin: 1ex 0em 1ex 0em ;
3568 }
3569
3570 div.oneauthor {
3571     display: inline-block ;
3572     margin: 0ex 1em 0ex 1em ;
3573 }
3574
3575 /*
3576 div.author table {
3577     margin: 1ex auto 0ex auto ;
3578     background: none ;
3579 }
3580
3581 div.author table tbody tr td { padding: .25ex ; }
3582 */
3583
3584 span.affiliation {font-size: .85em ; font-variant: small-caps; }
3585
3586 div.titledate {
3587     text-align: center ;
```

```
3588     font-size: .85em ;
3589     font-style: italic;
3590     margin: 1ex 0em 1ex 0em ;
3591 }
3592
3593
3594 nav.topnavigation{
3595     text-align: left ;
3596     padding: 0.5ex 1em 0.5ex 1em ;
3597 /*     margin: 2ex 0em 3ex 0em ; */
3598     margin: 0 ;
3599     border-bottom: 1px solid silver ;
3600     border-top: 1px solid silver ;
3601     clear:both ;
3602 }
3603
3604 nav.botnavigation{
3605     text-align: left ;
3606     padding: 0.5ex 1em 0.5ex 1em ;
3607 /*     margin: 3ex 0em 2ex 0em ; */
3608     margin: 0 ;
3609     border-top: 1px solid silver ;
3610     border-bottom: 1px solid silver ;
3611     clear:both ;
3612 }
3613
3614
3615 header {
3616     line-height: 1.2 ;
3617     font-size: 1em ;
3618     border-bottom: 1px solid silver ;
3619     margin: 0px ;
3620     padding: 2ex 1em 2ex 1em ;
3621     text-align:left ;
3622 }
3623
3624
3625 footer {
3626     font-size: .85em ;
3627     line-height: 1.2 ;
3628     margin-top: 1ex ;
3629     border-top: 1px solid silver ;
3630     padding: 2ex 1em 2ex 1em ;
3631     clear:both ;
3632     text-align:left ;
3633 }
3634
3635
3636 /* for \LinkHome, \LinkPrevious, and \LinkNext: */
3637 a.linkhome { font-weight:bold ; font-size: 1em ;}
3638
3639
3640 div.lateximagesource { padding: 0px ; margin: 0px ; display: none; }
3641
3642 img.lateximage{
3643     padding: 0pt ;
3644     margin: 0pt ;
3645     box-shadow: none ;
3646     border: none ;
3647     background: none ;
```

```
3648     max-width: 100% ;
3649     border-radius: 0ex ;
3650     border: none ;
3651 }
3652
3653
3654 div.sidetoccontainer {
3655     font-family: "DejaVu Serif", "Bitstream Vera Serif",
3656         "Lucida Bright", Georgia, serif;
3657     float: left ;
3658     width: 19%; /* room for border-right next to 80% main */
3659     margin: 0pt 0em 3ex 0pt ;
3660     border-right: 1px solid silver;
3661     border-bottom: 1px solid silver;
3662     background: #FAF7F4 ;
3663     font-size:.9em ;
3664     border-radius: 0px 0px 20px 0px ;
3665 }
3666
3667 div.sidetoccontents {
3668     overflow-y: auto ;
3669     width: 100% ;
3670     text-align: left ;
3671 }
3672
3673
3674 nav.sidetoc p {line-height:1.2 ; margin: 1ex .5em 1ex .5em ;
3675     text-indent: 0 ; }
3676
3677 nav.sidetoc p a {color:black ; font-size: .7em ;}
3678
3679 div.sidetoctitle {font-size: 1.2em; font-weight:bold; text-align:center;
3680     border-bottom: 1px solid silver ;    }
3681
3682 nav.sidetoc a:hover {text-decoration: underline ; }
3683
3684
3685
3686 section.textbody { margin: 0ex 1em 0ex 1em ;}
3687
3688
3689 div.multicolsheading { -webkit-column-span: all;
3690     -moz-column-span: all; column-span: all; }
3691 div.multicols {
3692     -webkit-columns: 3 auto ;
3693     -moz-columns: 3 auto ;
3694     columns: 3 auto ;
3695 }
3696 div.multicols p {margin-top: 0ex}
3697
3698
3699 /* Used for xfrac and nicefrac: */
3700 span.numerator {
3701     font-size: 60% ;
3702     vertical-align: .4em ;
3703 }
3704
3705 span.denominator {
3706     font-size: 60%
3707 }
```

```
3708
3709
3710 /* Used for algorithm2e: */
3711 div.alg2evline{
3712     margin-left: 1em ;
3713     padding-left: 1em ;
3714     border-left: 1px solid black ;
3715     border-radius: 0px 0px 0px 1ex ;
3716 }
3717
3718 div.alg2evsline{
3719     margin-left: 1em ;
3720     padding-left: 1em ;
3721     border-left: 1px solid black ;
3722 }
3723
3724 div.alg2enoline{
3725     margin-left: 1em ;
3726     padding-left: 1em ;
3727 }
3728
3729 span.alg2elinenumber{
3730     margin-right: .5em ;
3731     font-size: 60% ;
3732     color: red ;
3733 }
3734
3735
3736 /* Used for algorithmicx: */
3737 span.floatright { float: right ; }
3738
3739
3740 /* keyfloat and tocdata: */
3741 .floatnotes {
3742     margin: 0ex 5% 0ex 5% ;
3743     padding: 0ex 1em 0ex 1em ;
3744     font-size:.80em ;
3745     text-align: left ;
3746 }
3747
3748 .authorartist{
3749     display:block ;
3750     font-size:.70em ;
3751     font-style: italic;
3752 }
3753
3754 nav .authorartist{ display:inline; }
3755
3756
3757
3758 /* Native LaTeX theorems: */
3759
3760 .theoremcontents {
3761     font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ;
3762 }
3763
3764 .theoremlabel {
3765     font-style: normal; font-weight: bold ; margin-right: .5em ;
3766 }
3767
```

```
3768
3769
3770 /* theorem, amsthm, and ntheorem packages */
3771
3772 span.theoremheader,
3773 span.theoremheaderplain,
3774 span.theoremheaderdefinition,
3775 span.theoremheaderbreak,
3776 span.theoremheadermarginbreak,
3777 span.theoremheaderchangebreak,
3778 span.theoremheaderchange,
3779 span.theoremheadermargin
3780 {
3781     font-style:normal ; font-weight: bold ; margin-right: 1em ;
3782 }
3783
3784 span.amsthmnameplain,
3785 span.amsthmnamedefinition,
3786 span.amsthmnumberplain,
3787 span.amsthmnumberdefinition
3788 {
3789     font-style:normal ; font-weight: bold ;
3790 }
3791
3792
3793 span.amsthmnameremark,
3794 span.amsthmnumberremark
3795 {font-style:italic ; font-weight: normal ; }
3796
3797
3798 span.amsthmnoteplain,
3799 span.amsthmnotedefinition
3800 {font-style:normal ;}
3801
3802
3803 span.theoremheaderremark,
3804 span.theoremheaderproof,
3805 span.amsthmproofname
3806 {font-style:italic ; font-weight: normal ; margin-right: 1em ; }
3807
3808 span.theoremheadersc
3809 {
3810     font-style:normal ;
3811     font-variant: small-caps ;
3812     font-weight: normal ;
3813     margin-right: 1em ;
3814 }
3815
3816 .theoremendmark {float:right}
3817
3818 div.amsthmbodyplain, div.theorembodyplain, div.theorembodynonumberplain,
3819 div.theorembodybreak, div.theorembodynonumberbreak,
3820 div.theorembodymarginbreak,
3821 div.theorembodychangebreak,
3822 div.theorembodychange,
3823 div.theorembodymargin
3824 {
3825     font-style:italic;
3826     margin-top: 3ex ; margin-bottom: 3ex ;
3827 }
```

```
3828
3829 div.theorembodydefinition, div.theorembodyremark, div.theorembodyproof,
3830 div.theorembodyplainupright, nonumberplainuprightsc,
3831 div.amsthmbodydefinition, div.amsthmbodyremark,
3832 div.amsthmproof
3833 {
3834     font-style: normal ;
3835     margin-top: 3ex ; margin-bottom: 3ex ;
3836 }
3837
3838 span.amsthmnoteremark {}
3839
3840
3841 /* thmbox */
3842
3843 .thmbox {
3844     font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ;
3845     border: 1px solid gray ;
3846     padding: 1ex ;
3847 }
3848
3849 .thmboxtitle {
3850     font-style: normal; font-weight: bold ; margin-right: .5em ;
3851     border-bottom: 1px solid gray ;
3852 }
3853
3854 span.thmboxproofname, span.thmboxexamplename {
3855     font-weight: bold ;
3856 }
3857
3858 div.thmboxproof, div.thmboxexample {
3859     font-size: 0.85em ;
3860     margin: 2ex ;
3861 }
3862
3863 div.thmboxleftbar {
3864     border-left: 2px solid black ;
3865     padding-left: 1em ;
3866 }
3867
3868
3869
3870 /* For the backnaur package: */
3871 div.backnaur {
3872     display: block ;
3873     margin: 2ex 2em 2ex 2em ;
3874 }
3875
3876 div.backnaur p {
3877     margin: .25ex 0ex .25ex 0ex ;
3878 }
3879
3880 div.backnaurprod {
3881     display: inline-block ;
3882     min-width: 8em ;
3883     text-align:right ;
3884 }
3885
3886 div.backnaurdesc {
3887     display: inline-block ;
```

```
}


/* For the notes package: */
div.notesimportantnote, div.noteswarningnote, div.notesinformationnote {
    clear: both ;
    margin: 2ex 2em 2ex 2em ;
    border: 1px solid silver ;
}

div.notesicon {
    float:left ;
    display: inline-block ;
    background: gold ;
    padding: 0ex 1em 0ex 1em ;
    margin-right: 1em ;
    font-weight: bold ;
}

div.notescontents { font-style: italic }


/* nolbreaks package: */
span.nolbreaks { white-space: nowrap ; }


/*
For CSS LaTeX and related logos:
Based on spacing demonstrated by the metafont package.

The subscripts are shrunk instead of lowered below the baseline,
to avoid browser rendering errors with the line height in lists, etc.
*/

.latexlogofont {
    font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    font-variant: normal ;
}

.latexlogo {
    font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
}

.latexlogosup {
  text-transform: uppercase;
  letter-spacing: .03em ;
  font-size: 0.7em;
  vertical-align: 0.25em;
  margin-left: -0.4em;
  margin-right: -0.15em;
}

.latexlogosub {
  text-transform: uppercase;
/*  vertical-align: -0.27ex; */
  margin-left: -0.08em;
  margin-right: -0.07em;
/*  font-size: 1em; */
```

```
3948     font-size: .7em ;
3949 }
3950
3951 .latexlogotwoe {
3952   text-transform: none ;
3953   font-variant-numeric: oldstyle-nums ;
3954 }
3955
3956 .latexlogotwoesub {
3957   font-style:italic ;
3958 /*  vertical-align: -0.27ex; */
3959   margin-left: -0.11em;
3960   margin-right: -0.1em;
3961 /*  font-size: 1em; */
3962     font-size: .7em ;
3963 }
3964
3965 .xelatexlogo {
3966     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3967         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3968     letter-spacing: .03em ;
3969 }
3970
3971 .xelatexlogosub {
3972 /*  vertical-align: -0.27ex; */
3973   margin-left: -0.0667em;
3974   margin-right: -.05em;
3975 /*  font-size: 1em; */
3976     font-size: .7em ;
3977   letter-spacing: .03em ;
3978 }
3979
3980 .amslogo {
3981     font-family: "TeXGyreChorus","URW Chancery L",
3982         "Apple Chancery","ITC Zapf Chancery","Monotype Corsiva",
3983         "Linux Libertine O", "Nimbus Roman No 9 L",  "FreeSerif",
3984         "Hoefler Text", Times, "Times New Roman", serif ;
3985     font-style: italic ;
3986 }
3987
3988 .lyxlogo {
3989     font-family: "URW Classico", Optima, "Linux Biolinum O",
3990         "DejaVu Sans", "Bitstream Vera Sans", Geneva,
3991         Verdana, sans-serif ;
3992 }
3993
3994
3995 /* Only display top and bottom navigation if a small screen: */
3996 /* Hide the sidetoc if a small screen: */
3997 nav.topnavigation { display:none; }
3998 nav.botnavigation { display:none; }
3999
4000 /* Only display the sidetoc's webpage title if a small screen */
4001 span.sidetocthetitle { display: none }
4002
4003 @media screen and (max-width: 100em) {
4004     div.multicols {
4005         -webkit-columns: 2 auto ;
4006         -moz-columns: 2 auto ;
4007         columns: 2 auto ;
```

```
4008     }
4009 }
4010
4011 @media screen and (max-width: 50em) {
4012     div.sidetoccontainer {
4013         float: none ;
4014         width: 100% ;
4015         padding: 0 ;
4016         border-radius: 0 ;
4017         border-bottom: 1px solid black ;
4018         border-top: 1px solid black ;
4019         box-shadow: none ;
4020     }
4021     span.sidetocthetitle { display: inline }
4022     nav.topnavigation { display:block }
4023     nav.botnavigation { display:block }
4024     main.bodycontainer { width: 100% }
4025     .marginpar {
4026         max-width: 100%;
4027         float: none;
4028         display:block ;
4029         margin: 1ex 1em 1ex 1em ;
4030     }
4031     div.multicols {
4032         -webkit-columns: 1 auto ;
4033         -moz-columns: 1 auto ;
4034         columns: 1 auto ;
4035     }
4036 }
4037
4038 @media print {
4039     body {
4040         font-family: "Linux Libertine O",
4041         "DejaVu Serif", "Bitstream Vera Serif",
4042         "Liberation Serif", "Nimbus Roman No 9 L",
4043         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4044     }
4045     div.sidetoccontainer { display:none; }
4046     nav.topnavigation { display: none; }
4047     nav.botnavigation { display: none; }
4048     main.bodycontainer { width: 100% }
4049 }
4050
4051 @media handheld {
4052     div.sidetoccontainer { display:none; }
4053     nav.topnavigation { display:block }
4054     nav.botnavigation { display:block }
4055     main.bodycontainer { width: 100% }
4056 }
4057
4058 @media projection {
4059     div.sidetoccontainer { display:none; }
4060     nav.topnavigation { display:block }
4061     nav.botnavigation { display:block }
4062     main.bodycontainer { width: 100% }
4063 }
4064 \end{filecontents*}
4065 % \end{Verbatim}% for syntax highlighting
4066 \end{LWRwriteconf}
```

40.5 lwarp_sagebrush.css

lwarp_sagebrush.css (*file*) An optional CSS which may be used for a semi-modern appearance.

If used, this must be present both when compiling the project and also when distributing the HTML files.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{lwarp_sagebrush.css}
@import url("lwarp.css") ;


A:link {color:#105030 ; text-decoration: none ; }
A:visited {color:#705030 ; text-shadow:1px 1px 2px #a0a0a0;}
A:hover {color:#006000 ; text-decoration: underline ; text-shadow:0px 0px 2px #a0a0a0;}
A:active {color:#00C000 ; text-shadow:1px 1px 2px #a0a0a0;}



div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
{
    font-family: "URW Classico", Optima, "Linux Biolinum O",
        "Linux Libertine O", "Liberation Serif",
        "Nimbus Roman No 9 L",  "FreeSerif",
        "Hoefler Text", Times, "Times New Roman", serif;
    font-variant: small-caps ;
    font-weight: normal ;
    color: #304070 ;
    text-shadow: 2px 2px 3px #808080;
}

h1 {    /* title of the entire website, used on each page */
    font-variant: small-caps ;
    color: #304070 ;
    text-shadow: 2px 2px 3px #808080;
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C4);
}

h1 {
  border-bottom: 1px solid #304070;
/*  border-top: 2px solid #304070; */
}

h2 {
  border-bottom: 1px solid #304070;
/*  border-top: 2px solid #304070; */
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to bottom, #F7F7F0, #DAD0C0);
}



div.abstract {
    background: #f5f5eb ;
    background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);

  border: 1px solid silver;
    border-radius: 1em ;
}
```

```
4120
4121 div.abstract dl {line-height:1.5;}
4122 div.abstract dt {color:#304070;}
4123
4124 div.abstracttitle{
4125     font-family: "URW Classico", Optima, "Linux Biolinum O",
4126         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4127         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4128     font-weight:bold;
4129     font-variant: small-caps ;
4130     font-size:1.5em;
4131     border-bottom: 1px solid silver ;
4132     color: #304070 ;
4133     text-align: center ;
4134     text-shadow: 1px 1px 2px #808080;
4135 }
4136
4137 span.abstractrunintitle{
4138     font-family: "URW Classico", Optima, "Linux Biolinum O",
4139         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4140         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4141     font-weight:bold;
4142 }
4143
4144
4145 div.epigraph, div.dictum {
4146     background: #f5f5eb ;
4147     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4148
4149     border: 1px solid silver ;
4150     border-radius: 1ex ;
4151     box-shadow: 3px 3px 3px #808080 ;
4152 }
4153
4154
4155 .example {
4156     background-color: #f5f5eb ;
4157     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4158
4159 }
4160
4161 div.exampletitle{
4162     font-family: "URW Classico", Optima, "Linux Biolinum O",
4163         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4164         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4165     font-weight:bold;
4166     font-variant: small-caps ;
4167     border-bottom: 1px solid silver ;
4168     color: #304070 ;
4169     text-align: center ;
4170     text-shadow: 1px 1px 2px #808080;
4171 }
4172
4173
4174 .sidebar {
4175     background-color: #f5f5eb ;
4176     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4177
4178 }
4179
```

```
4180 div.sidebartitle{
4181     font-family: "URW Classico", Optima, "Linux Biolinum O",
4182         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4183         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4184     font-weight:bold;
4185     font-variant: small-caps ;
4186     border-bottom: 1px solid silver ;
4187     color: #304070 ;
4188     text-align: center ;
4189     text-shadow: 1px 1px 2px #808080;
4190 }
4191
4192
4193 .fancyvrblabel {
4194     font-family: "URW Classico", Optima, "Linux Biolinum O",
4195         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4196         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4197     font-weight:bold;
4198     font-variant: small-caps ;
4199     font-size: 1.5em ;
4200     color: #304070 ;
4201     text-align: center ;
4202     text-shadow: 1px 1px 2px #808080;
4203 }
4204
4205 div.minipage {
4206     background-color: #eeeee7 ;
4207     border: 1px solid silver ;
4208     border-radius: 1ex ;
4209 }
4210
4211 table div.minipage { background: none ; border: none ; }
4212
4213 div.framebox div.minipage {border:none ; background:none}
4214
4215 section.textbody > div.minipage {
4216     box-shadow: 3px 3px 3px #808080 ;
4217 }
4218
4219 div.fboxBlock div.minipage { box-shadow: none ; }
4220
4221 .framed .minipage , .framedleftbar .minipage {
4222     border: none ;
4223     background: none ;
4224     padding: 0ex ;
4225     margin: 0ex ;
4226 }
4227
4228 figure.figure .minipage, div.figurecaption .minipage { border: none; }
4229
4230 div.marginblock div.minipage ,
4231 div.marginparblock div.minipage
4232     { border: none; }
4233
4234 figure , div.marginblock {
4235     background-color: #eeeee7 ;
4236     border: 1px solid silver ;
4237     border-radius: 1ex ;
4238     box-shadow: 3px 3px 3px #808080 ;
4239 }
```

```
4240
4241 figure figure {
4242     border: 1px solid silver ;
4243     margin: 0em ;
4244     box-shadow: none ;
4245 }
4246
4247 /*
4248 div.figurecaption {
4249     border-top: 1px solid silver ;
4250     border-bottom: 1px solid silver ;
4251     background-color: #e8e8e8 ;
4252 }
4253 */
4254
4255
4256 div.table {
4257     box-shadow: 3px 3px 3px #808080 ;
4258 }
4259
4260 /*
4261 .tnotes {
4262     background: #e8e8e8;
4263     border: 1px solid silver;
4264 }
4265 */
4266
4267
4268 nav.topnavigation{
4269     background-color: #b0b8b0 ;
4270     background-image: linear-gradient(to bottom,#e0e0e0,#b0b8b0) ;
4271 }
4272
4273 nav.botnavigation{
4274     background-color: #b0b8b0 ;
4275     background-image: linear-gradient(to top,#e0e0e0,#b0b8b0) ;
4276 }
4277
4278
4279
4280 header{
4281     background-color: #F7F7F0 ;
4282     background-image: linear-gradient(to top, #F7F7F0, #b0b8b0);
4283 }
4284
4285 footer{
4286     background-color: #F7F7F0 ;
4287     background-image: linear-gradient(to bottom, #F7F7F0, #b0b8b0);
4288 }
4289
4290
4291
4292 div.sidetoccontainer {
4293     background-color: #F7F7F0 ;
4294     background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C0);
4295     box-shadow: 3px 3px 3px #808080 ;
4296     }
4297
4298 div.sidetoctitle {color: #304070 ; }
4299
```

```
nav.sidetoc a:hover {
    color:#006000 ;
    text-decoration: none ;
    text-shadow:0px 0px 2px #a0a0a0;
}


@media screen and (max-width: 45em) {
    div.sidetoccontainer { border-radius: 0 ; }
}


\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.6 lwarp_formal.css

`lwarp_formal.css` (*file*) An optional css which may be used for a more formal appearance.

If used, this must be present both when compiling the project and also when distributing the HTML files.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{lwarp_formal.css}
@import url("lwarp.css") ;



A:link {color:#802020 ; text-decoration:none; }
A:visited {color:#802020 ; text-shadow:none ;}
A:hover {color:#400000 ; text-shadow:none ;}
A:active {color:#C00000 ; text-shadow:none ;}


body {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    background: #fffcf5;
}

span.textrm {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
}

span.textsf {
     font-family: "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
}



div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
```

```
{
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    color: #800000 ;
    text-shadow: none ;
}

h1, h2 {
    background-color: #fffcf5 ;
    background-image: none ;
    border-bottom: 1px solid #808080;
/*    border-top: 2px solid #808080; */
}

div.abstracttitle {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    color: black ;
    text-shadow: none ;
}

span.abstractrunintitle {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    color: black ;
    text-shadow: none ;
}

div.abstract { font-size: 100% }

.sidebar {
    background: #fffcf5;
    background-image: none ;
  margin: 2em 5% 2em 5%;
  padding: 0.5em 1em;
  border: none ;
  border-top : 1px solid silver;
  border-bottom : 1px solid silver;
  font-size: 90% ;
}

div.sidebartitle{
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    color: #800000 ;
    text-shadow: none ;
    border: none ;
}

.example {
    background: #fffcf5;
    background-image: none ;
```

```
  margin: 2em 5% 2em 5%;
  padding: 0.5em 1em;
  border: none ;
  border-top : 1px solid silver;
  border-bottom : 1px solid silver;
}

div.exampletitle{
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    color: #800000 ;
    text-shadow: none ;
    border: none ;
}

div.fancyvrblabel{
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    color: #800000 ;
    text-shadow: none ;
    border: none ;
}



figure {
    margin: 5ex 5% 5ex 5% ;
    padding: 1ex 1em 1ex 1em ;
    background-color: #fffcf5 ;
    overflow-x: auto ;
    border: none ;
/*     border-top: 1px solid silver; */
/*     border-bottom: 1px solid silver; */
}


div.figurecaption , .lstlisting {
    border: none ;
/*     border-top: 1px solid silver ; */
/*     border-bottom: 1px solid silver ; */
    background-color: #fffcf5 ;
}

.tnotes {
    background: #fffcf5 ;
    border-top: 1px solid silver ;
    border-bottom: 1px solid silver ;
}

.theorem {
        background: none ;
}

.minipage {
    background-color: #fffcf5 ;
    border: none ;
```

```
}

div.floatrow figure { border: none ; }

figure figure { border: none ; }


nav.toc, nav.lof, nav.lot, nav.lol {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
}

div.sidetoccontainer {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    background-image: linear-gradient(to bottom, #fffcf5, #C0C0C0);
}

div.sidetoctitle{
    color: #800000 ;
}

header{
    background-color: #e0e0e0 ;
    background-image: linear-gradient(to top, #fffcf5, #b0b0b0);
    text-align:center ;
}

footer{
    background-color: #e0e0e0 ;
    background-image: linear-gradient(to bottom, #fffcf5, #b0b0b0);
    padding: 2ex 1em 2ex 1em ;
    text-align:left ;
}

nav.botnavigation {
    background: #dedcd5 ;
    border-top: 1px solid black ;
}
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.7 sample_project.css

sample_project.css (*file*) The project-specific CSS file. Use with \CSSFilename.

If used, this must be present both when compiling the project and also when distributing the HTML files.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{sample_project.css}
/* ( --- Start of project.css --- ) */
/* ( --- A sample project-specific CSS file for lwarp --- ) */
```

```
4520
4521 /* Uncomment one of the following: */
4522 @import url("lwarp.css") ;
4523 /* @import url("lwarp_formal.css") ; */
4524 /* @import url("lwarp_sagebrush.css") ; */
4525
4526 /* Project-specific CSS setting follow here. */
4527 /* . . . */
4528
4529 /* ( --- End of project.css --- ) */
4530 \end{filecontents*}
4531 % \end{Verbatim}% for syntax highlighting
4532 \end{LWRwriteconf}
```

40.8 lwarp.ist

`lwarp.ist` (*file*) Used to modify the index for lwarp.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

The page compositor line is for memoir's `\specialindex`.

Config file:

```
4533 \begin{LWRwriteconf}
4534 \begin{filecontents*}[overwrite]{lwarp.ist}
4535 preamble
4536 "\\begin{theindex}
4537   \\providecommand*\\lettergroupDefault[1]{}
4538   \\providecommand*\\lettergroup[1]{%
4539       \\par\\textbf{#1}\\par
4540       \\nopagebreak
4541   }
4542 "
4543 headings_flag 1
4544 heading_prefix "
4545   \\lettergroup{"
4546 heading_suffix "}"
4547 delim_0 ", \\hyperindexref{"
4548 delim_1 ", \\hyperindexref{"
4549 delim_2 ", \\hyperindexref{"
4550 delim_n "}, \\hyperindexref{"
4551 delim_r "} -- \\hyperindexref{"
4552 delim_t "}"
4553 page_compositor "."
4554 \end{filecontents*}
4555 % \end{Verbatim}% for syntax highlighting
4556 \end{LWRwriteconf}
```

40.9 lwarp.xdy

`lwarp.xdy` (*file*) Used to modify the index for lwarp.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

See:
https://tex.stackexchange.com/questions/80300/how-can-i-convince-hyperref-and-xindy-to-play-together-nicely

Config file:
```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{lwarp.xdy}
(require "tex/inputenc/latin.xdy")
(merge-rule "\\PS *" "Postscript")
(require "texindy.xdy")
(require "page-ranges.xdy")
(require "book-order.xdy")
(define-location-class "arabic-page-numbers"
    ("arabic-numbers") :min-range-length 1)
(require "makeindex.xdy")
(define-attributes (("hyperindexref")))
(markup-locref :open "\hyperindexref{" :close "}")
(markup-locref :open "\hyperindexref{" :close "}" :attr "hyperpage")
(markup-locref :open "\textbf{\hyperindexref{" :close "}}" :attr "textbf")
(markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
(define-location-class-order ("roman-page-numbers"
                  "arabic-page-numbers"
                  "alpha-page-numbers"
                  "Roman-page-numbers"
                  "Alpha-page-numbers"
                  "see"
                  "seealso"))
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.10 lwarp_one_limage.cmd

lwarp_one_limage.cmd (*file*) Used by lwarp to help make lateximages when using WINDOWS.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

The arguments are each of the three fields from <project>-images.txt, and also the base name of the source file.

MiKTeX does not allow file lwarp_one_limage.cmd to be created directly by *lwarpmk*, so lwarp_one_limage.txt is created instead, then copied to lwarp_one_limage.cmd by *lwarpmk*. This occurs each time *lwarpmk* used to create lateximages.

Config file:
```
\begin{LWRwriteconf}
\immediate\openout\LWR@quickfile=lwarp_one_limage.txt
\immediate\write\LWR@quickfile{%
    pdfseparate -f \LWRpercent 1 -l \LWRpercent 1 \LWRpercent 4_html.pdf %
   \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent\LWRpercent d.pdf%
}
\immediate\write\LWR@quickfile{%
    pdfcrop --hires  --margins \LWRopquote0 1 0 0\LWRopquote\space %
    \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf %
    \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
}
\immediate\write\LWR@quickfile{%
   pdftocairo -svg -noshrink \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf %
    \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.svg%
```

```
}
\immediate\write\LWR@quickfile{%
    del \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
}
\immediate\write\LWR@quickfile{%
    del \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf%
}
\immediate\write\LWR@quickfile{exit}
\immediate\closeout\LWR@quickfile
\end{LWRwriteconf}
```

40.11 lwarp_mathjax.txt

(*Emulates or patches code by* Davide P. Cervone.)

lwarp_mathjax.txt (*file*) The default MathJax script used by lwarp when using MathJax. A recent version of MathJax is used, as served by the recommended repository. Adjustments are made to allow LaTeX to control the equation tags and provide for starred macros.

\MathJaxFilename determines which script file is copied into the HTML pages, and defaults to lwarp_mathjax.txt. The script files must be present when compiling the project, but do not need to be present when distributing the resulting HTML files.

custom script To generate a custom script, such as to use a local repository, copy lwarp_mathjax.txt to a new file, make changes while keeping lwarp's adjustments for equation numbering and starred macros, and use \MathJaxFilename to select the new filename.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{lwarp_mathjax.txt}
<script>
// Lwarp MathJax emulation code
//
// Based on code by Davide P. Cervone.
// Equation numbering: https://github.com/mathjax/MathJax/issues/2427
// Starred and ifnextchar macros: https://github.com/mathjax/MathJax/issues/2428
// \left, \right delimiters: https://github.com/mathjax/MathJax/issues/2535
//
// Modified by Brian Dunn to adjust equation numbering and add subequations.
//
// LaTeX can use \seteqnumber{subequations?}{section}{number} before each equation.
// subequations? is 0 usually, 1 if inside subequations.
// section is a string printed as-is, or empty.
// number is auto-incremented by MathJax between equations.
//
MathJax = {
  subequations: "0",
  section: "",
  loader: {
    load: ['[tex]/tagformat', '[tex]/textmacros'],
  },
  startup: {
    ready() {
      //  These would be replaced by import commands if you wanted to make
      //  a proper extension.
      const Configuration = MathJax._.input.tex.Configuration.Configuration;
      const CommandMap = MathJax._.input.tex.SymbolMap.CommandMap;
      const Macro = MathJax._.input.tex.Symbol.Macro;
```

```
4636      const TexError = MathJax._.input.tex.TexError.default;
4637      const ParseUtil = MathJax._.input.tex.ParseUtil.default;
4638      const expandable = MathJax._.util.Options.expandable;
4639
4640      //  Insert the replacement string into the TeX string, and check
4641      //  that there haven't been too many maxro substitutions (prevents
4642      //  infinite loops).
4643      const useArgument = (parser, text) => {
4644     parser.string = ParseUtil.addArgs(parser, text, parser.string.slice(parser.i));
4645        parser.i = 0;
4646        if (++parser.macroCount > parser.configuration.options.maxMacros) {
4647          throw new TexError('MaxMacroSub1',
4648          'MathJax maximum macro substitution count exceeded; ' +
4649          'is there a recursive macro call?');
4650        }
4651      }
4652
4653      //  Create the command map for:
4654      //    \ifstar, \ifnextchar, \ifblank, \ifstrequal, \gsub, \seteqnumber
4655      new CommandMap('Lwarp-macros', {
4656        ifstar: 'IfstarFunction',
4657        ifnextchar: 'IfnextcharFunction',
4658        ifblank: 'IfblankFunction',
4659        ifstrequal: 'IfstrequalFunction',
4660        gsubstitute: 'GsubstituteFunction',
4661        seteqnumber: 'SeteqnumberFunction'
4662      }, {
4663        //  This function implements an ifstar macro.
4664        IfstarFunction(parser, name) {
4665          const resultstar = parser.GetArgument(name);
4666          const resultnostar = parser.GetArgument(name);
4667         const star = parser.GetStar();               // true if there is a *
4668          useArgument(parser, star ? resultstar : resultnostar);
4669        },
4670
4671        //  This function implements an ifnextchar macro.
4672        IfnextcharFunction(parser, name) {
4673          let whichchar = parser.GetArgument(name);
4674          if (whichchar.match(/^(?:0x[0-9A-F]+|[0-9]+)$/i)) {
4675            // $ syntax highlighting
4676            whichchar = String.fromCodePoint(parseInt(whichchar));
4677          }
4678          const resultnextchar = parser.GetArgument(name);
4679          const resultnotnextchar = parser.GetArgument(name);
4680          const gotchar = (parser.GetNext() === whichchar);
4681         useArgument(parser, gotchar ? resultnextchar : resultnotnextchar);
4682        },
4683
4684        // This function implements an ifblank macro.
4685        IfblankFunction(parser, name) {
4686          const blankarg = parser.GetArgument(name);
4687          const resultblank = parser.GetArgument(name);
4688          const resultnotblank = parser.GetArgument(name);
4689          const isblank = (blankarg.trim() == "");
4690          useArgument(parser, isblank ? resultblank : resultnotblank);
4691        },
4692
4693        // This function implements an ifstrequal macro.
4694        IfstrequalFunction(parser, name) {
4695          const strequalfirst = parser.GetArgument(name);
```

```
4696          const strequalsecond = parser.GetArgument(name);
4697          const resultequal = parser.GetArgument(name);
4698          const resultnotequal = parser.GetArgument(name);
4699          const isequal = (strequalfirst == strequalsecond);
4700          useArgument(parser, isequal ? resultequal : resultnotequal);
4701        },
4702
4703        // This function implements a gsub macro.
4704        GsubstituteFunction(parser, name) {
4705          const gsubfirst = parser.GetArgument(name);
4706          const gsubsecond = parser.GetArgument(name);
4707          const gsubthird = parser.GetArgument(name);
4708          let gsubresult=gsubfirst.replace(gsubsecond, gsubthird);
4709          useArgument(parser, gsubresult);
4710        },
4711
4712        //  This function modifies the equation numbers.
4713        SeteqnumberFunction(parser, name) {
4714            //  Get the macro parameters
4715          const star = parser.GetStar();               // true if there is a *
4716        const optBrackets = parser.GetBrackets(name);  // contents of optional brackets
4717        const newsubequations = parser.GetArgument(name);  // the subequations argument
4718        const neweqsection = parser.GetArgument(name);  // the eq section argument
4719        const neweqnumber = parser.GetArgument(name);  // the eq number argument
4720        MathJax.config.subequations=newsubequations ;  // a string with boolean meaning
4721        MathJax.config.section=neweqsection ;       // a string with numeric meaning
4722            parser.tags.counter = parser.tags.allCounter = neweqnumber ;
4723        }
4724
4725      });
4726
4727      //  Create the Lwarp-macros package
4728      Configuration.create('Lwarp-macros', {
4729        handler: {macro: ['Lwarp-macros']}
4730      });
4731
4732      MathJax.startup.defaultReady();
4733
4734      // For forward references:
4735      MathJax.startup.input[0].preFilters.add(({math}) => {
4736        if (math.inputData.recompile){
4737        MathJax.config.subequations = math.inputData.recompile.subequations;
4738            MathJax.config.section = math.inputData.recompile.section;
4739        }
4740      });
4741      MathJax.startup.input[0].postFilters.add(({math}) => {
4742        if (math.inputData.recompile){
4743        math.inputData.recompile.subequations = MathJax.config.subequations;
4744            math.inputData.recompile.section = MathJax.config.section;
4745        }
4746      });
4747
4748        // For \left, \right with unicode-math:
4749        const {DelimiterMap} = MathJax._.input.tex.SymbolMap;
4750        const {Symbol} = MathJax._.input.tex.Symbol;
4751        const {MapHandler} = MathJax._.input.tex.MapHandler;
4752        const delimiter = MapHandler.getMap('delimiter');
4753        delimiter.add('\\lBrack', new Symbol('\\lBrack', '\u27E6'));
4754        delimiter.add('\\rBrack', new Symbol('\\rBrack', '\u27E7'));
4755        delimiter.add('\\lAngle', new Symbol('\\lAngle', '\u27EA'));
```

```
4756        delimiter.add('\\rAngle', new Symbol('\\rAngle', '\u27EB'));
4757        delimiter.add('\\lbrbrak', new Symbol('\\lbrbrak', '\u2772'));
4758        delimiter.add('\\rbrbrak', new Symbol('\\rbrbrak', '\u2773'));
4759        delimiter.add('\\lbag', new Symbol('\\lbag', '\u27C5'));
4760        delimiter.add('\\rbag', new Symbol('\\rbag', '\u27C6'));
4761      delimiter.add('\\llparenthesis', new Symbol('\\llparenthesis', '\u2987'));
4762      delimiter.add('\\rrparenthesis', new Symbol('\\rrparenthesis', '\u2988'));
4763        delimiter.add('\\llangle', new Symbol('\\llangle', '\u2989'));
4764        delimiter.add('\\rrangle', new Symbol('\\rrangle', '\u298A'));
4765        delimiter.add('\\Lbrbrak', new Symbol('\\Lbrbrak', '\u27EC'));
4766        delimiter.add('\\Rbrbrak', new Symbol('\\Rbrbrak', '\u27ED'));
4767        delimiter.add('\\lBrace', new Symbol('\\lBrace', '\u2983'));
4768        delimiter.add('\\rBrace', new Symbol('\\rBrace', '\u2984'));
4769        delimiter.add('\\lParen', new Symbol('\\lParen', '\u2985'));
4770        delimiter.add('\\rParen', new Symbol('\\rParen', '\u2986'));
4771        delimiter.add('\\lbrackubar', new Symbol('\\lbrackubar', '\u298B'));
4772        delimiter.add('\\rbrackubar', new Symbol('\\rbrackubar', '\u298C'));
4773      delimiter.add('\\lbrackultick', new Symbol('\\lbrackultick', '\u298D'));
4774      delimiter.add('\\rbracklrtick', new Symbol('\\rbracklrtick', '\u298E'));
4775      delimiter.add('\\lbracklltick', new Symbol('\\lbracklltick', '\u298F'));
4776      delimiter.add('\\rbrackurtick', new Symbol('\\rbrackurtick', '\u2990'));
4777        delimiter.add('\\langledot', new Symbol('\\langledot', '\u2991'));
4778        delimiter.add('\\rangledot', new Symbol('\\rangledot', '\u2992'));
4779        delimiter.add('\\lparenless', new Symbol('\\lparenless', '\u2993'));
4780        delimiter.add('\\rparengtr', new Symbol('\\rparengtr', '\u2994'));
4781        delimiter.add('\\Lparengtr', new Symbol('\\Lparengtr', '\u2995'));
4782        delimiter.add('\\Rparenless', new Symbol('\\Rparenless', '\u2996'));
4783        delimiter.add('\\lblkbrbrak', new Symbol('\\lblkbrbrak', '\u2997'));
4784        delimiter.add('\\rblkbrbrak', new Symbol('\\rblkbrbrak', '\u2998'));
4785        delimiter.add('\\lvzigzag', new Symbol('\\lvzigzag', '\u29D8'));
4786        delimiter.add('\\rvzigzag', new Symbol('\\rvzigzag', '\u29D9'));
4787        delimiter.add('\\Lvzigzag', new Symbol('\\Lvzigzag', '\u29DA'));
4788        delimiter.add('\\Rvzigzag', new Symbol('\\Rvzigzag', '\u29DB'));
4789       delimiter.add('\\lcurvyangle', new Symbol('\\lcurvyangle', '\u29FC'));
4790       delimiter.add('\\rcurvyangle', new Symbol('\\rcurvyangle', '\u29FD'));
4791        delimiter.add('\\Vvert', new Symbol('\\Vvert', '\u2980'));
4792    }   // ready
4793  },    // startup
4794
4795  tex: {
4796    packages: {'[+]': ['tagformat', 'Lwarp-macros', 'textmacros']},
4797    tags: "ams",
4798        tagformat: {
4799            number: function (n) {
4800                if(MathJax.config.subequations==0)
4801                    return(MathJax.config.section + n);
4802                else
4803                 return(MathJax.config.section + String.fromCharCode(96+n));
4804            },
4805        },
4806  }
4807 }
4808 </script>
4809
4810 <script
4811    id="MathJax-script"
4812    src="https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-svg.js"
4813 ></script>
4814 \end{filecontents*}
4815 % \end{Verbatim}% for syntax highlighting
```

```
\end{LWRwriteconf}
```

40.12 `lwarpmk.lua` — `lwarpmk` option

`lwarpmk` (*Opt*) Creates a local copy of *lwarpmk*.

`lwarpmk` (*Prog*) Command-line utility to process lwarp files and images.

parallel processing `lateximages` and SVG math images are generated using multiple processes in parallel. For UNIX and LINUX, every 32 images the `wait` command is issued to wait for the previous batch of images to finish processing before starting a new batch. For WINDOWS, every 32 images one task is dispatched with

```
START /B /WAIT /BELOWNORMAL
```

which causes the operating system to wait until this lesser-priority tasks finishes, hopefully also waiting for the normal priority tasks which were already in progress to also complete. Afterwards, the next batch of images is started.

The following is only generated if the `lwarpmk` option was given to lwarp.

```
\begin{LWRcreatelwarpmk}

\begin{filecontents*}[overwrite]{lwarpmk.lua}
#!/usr/bin/env texlua

-- Copyright 2016-2022 Brian Dunn


printversion = "v0.911"
requiredconfversion = "2" -- also at *lwarpmk.conf

function printhelp ()
print ("lwarpmk: Use lwarpmk -h or lwarpmk --help for help.") ;
end


function printusage ()
--
-- Print the usage of the lwarpmk command:
--
print ( [[

lwarpmk print [-p project]: Compile the print version if necessary.
lwarpmk print1 [-p project]: Forced single compile of the print version.
lwarpmk printindex [-p project]: Process print indexes.
lwarpmk printglossary [-p project]: Process the glossary for the print version.
lwarpmk html [-p project]: Compile the HTML version if necessary.
lwarpmk html1 [-p project]: Forced single compile of the HTML version.
lwarpmk htmlindex [-p project]: Process HTML indexes.
lwarpmk htmlglossary [-p project]: Process the glossary for the html version.
lwarpmk again [-p project]: Touch the source code to trigger recompiles.
lwarpmk limages [-p project]: Process the "lateximages" created by lwarp.sty.
lwarpmk pdftohtml [-p project]:
    For use with latexmk or a Makefile:
    Converts project_html.pdf to project_html.html and individual HTML files.
    Finishes the HTML conversion even if there was a compile error.
lwarpmk pdftosvg <list of file names>: Converts each PDF file to SVG.
```

```
lwarpmk epstopdf <list of file names>: Converts each EPS file to PDF.
lwarpmk clean [-p project]: Remove *.aux, *.toc, *.lof/t,
    *.idx, *.ind, *.bbl, *.log, *_html_inc.*, .gl*,
    *_html.pdf, *_html.html, *_html.sidetoc
lwarpmk cleanall [-p project]: Remove auxiliary files, project.pdf, *.html
lwarpmk cleanlimages: Removes all images from the "lateximages" directory.
lwarpmk -v: Print the version number.
lwarpmk -h: Print this help message.
lwarpmk --help: Print this help message.

]] )
-- printconf ()
end


function splitfilename ( pathandfilename )
--
-- Separates out the path and extension from a filename.
-- Returns path, filename with extension, and extension.
-- Ex: thispath, thisfilename, thisextension = splitfilename ("path/to/filename.ext")
--
-- https://www.fhug.org.uk/wiki/wiki/doku.php?id=plugins:code_snippets:
--      split_filename_in_to_path_filename_and_extension
--
    if lfs.attributes(pathandfilename,"mode") == "directory" then
      local strPath = pathandfilename:gsub("[\\/]$","") -- $ (syntax highlighting)
        return strPath.."\\","",""
    end
    pathandfilename = pathandfilename.."."
    return pathandfilename:match("^(.-)([^\\/]-)%.([^\\/%.]-)%.?$")
end


function splitfile (destfile,sourcefile)
--
-- Split one large sourcefile into a number of files,
-- starting with destfile.
-- The file is split at each occurance of <!--|Start file|newfilename|*
-- If lwarp is in use, sets usinglwarp.
--
usinglwarp = false ;
print ("lwarpmk: Splitting " .. sourcefile .. " into " .. destfile) ;
local sfile = io.open(sourcefile)
io.output(destfile)
for line in sfile:lines() do
i,j,copen,cstart,newfilename = string.find (line,"(.*)|(.*)|(.*)|") ;
if ( (i~= nil) and (copen == "<!--") and (cstart == "Start file")) then
    -- split the file
    io.output(newfilename) ;
else
if ( (i~= nil) and (copen == "<!--") and (cstart == "Using lwarp")) then
    -- verified the use of \usepackage{lwarp}
    usinglwarp = true ;
else
    -- not a splitpoint
    io.write (line .. "\n") ;
end end
end -- do
io.close(sfile)
if ( usinglwarp == false ) then
```

```
4913     print ("lwarpmk: ===")
4914     print ("lwarpmk: \\usepackage{lwarp} was not detected.")
4915     print ("lwarpmk: The HTML output will not be correct.")
4916     print ("lwarpmk: Ensured that \\usepackage{lwarp} is enabled,")
4917     print ("lwarpmk: then lwarpmk print and lwarpmk html again.")
4918     print ("lwarpmk: ===")
4919 end
4920 end -- function
4921
4922
4923 function cvalueerror ( line, linenum , cvalue )
4924 --
4925 -- Incorrect value, so print an error and exit.
4926 --
4927     print ("lwarpmk: ===")
4928     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
4929     print (
4930         "lwarpmk: incorrect variable value \"" .. cvalue ..
4931         "\" in lwarpmk.conf.\n"
4932     ) ;
4933     print ("lwarpmk: ===")
4934 --    printconf () ;
4935     os.exit(1) ;
4936 end
4937
4938
4939 function printhowtorecompile ()
4940 -- Tells the user how to recompile to regenerate the configuration files.
4941   print ("lwarpmk: The configuration files lwarpmk.conf and "..sourcename..".lwarpmkconf" )
4942     print ("lwarpmk:   must be updated.  To do so, recompile" )
4943     print ("lwarpmk:   " , sourcename..".tex" )
4944     if ( printlatexcmd == "" ) then
4945         print ("lwarpmk:   using xe/lua/pdflatex," )
4946     else
4947         print ("lwarpmk:   using the command:")
4948         print ("lwarpmk:   " , printlatexcmd )
4949     end
4950     print ("lwarpmk:   then use lwarpmk again.")
4951 end -- printhowtorecompile
4952
4953
4954 function ignoreconf ()
4955 -- Global argument index
4956 argindex = 2
4957 end
4958
4959 function loadconf ()
4960 --
4961 -- Load settings from the project's "lwarpmk.conf" file:
4962 --
4963 -- Default configuration filename:
4964 local conffile = "lwarpmk.conf"
4965 local confroot = "lwarpmk"
4966 -- Global argument index
4967 argindex = 2
4968 -- Optional configuration filename:
4969 if ( arg[argindex] == "-p" ) then
4970     argindex = argindex + 1
4971     confroot = arg[argindex]
4972     conffile = confroot..".lwarpmkconf"
```

```
4973     argindex = argindex + 1
4974 end
4975 -- Additional defaults:
4976 confversion = "0"
4977 opsystem = "Unix"
4978 imagesdirectory = "lateximages"
4979 imagesname = "image-"
4980 latexmk = "false"
4981 printlatexcmd = ""
4982 HTMLlatexcmd = ""
4983 printindexcmd = ""
4984 HTMLindexcmd = ""
4985 latexmkindexcmd = ""
4986 -- to be removed:
4987 -- indexprog = "makeindex"
4988 -- makeindexstyle = "lwarp.ist"
4989 -- xindylanguage = "english"
4990 -- xindycodepage = "utf8"
4991 -- xindystyle = "lwarp.xdy"
4992 -- pdftotextenc = "UTF-8"
4993 glossarycmd = "makeglossaries"
4994 -- Verify the file exists:
4995 if (lfs.attributes(conffile,"mode")==nil) then
4996     -- file not exists
4997     print ("lwarpmk: ===")
4998     print ("lwarpmk: File \"" .. conffile .."\" does not exist.")
4999     print ("lwarpmk: Move to the project's source directory,")
5000     print ("lwarpmk: recompile using pdflatex, xelatex, or lualatex,")
5001     print ("lwarpmk: then try using lwarpmk again.")
5002     if ( arg[argindex] ~= nil ) then
5003         print (
5004             "lwarpmk: (\"" .. confroot ..
5005             "\" does not appear to be a project name.)"
5006         )
5007     end
5008     print ("lwarpmk: ===")
5009     printhelp () ;
5010     os.exit(1) -- exit the entire lwarpmk script
5011 else -- file exists
5012 -- Read the file:
5013 print ("lwarpmk: Reading " .. conffile ..".")
5014 local cfile = io.open(conffile)
5015 -- Scan each line, parsing each line as: name = [[string]]
5016 local linenum = 0
5017 for line in cfile:lines() do -- scan lines
5018 linenum = linenum + 1
5019 i,j,cvarname,cvalue = string.find (line,"([%w-_]*)%s*=%s*%[%[([^%]]*)%]%]") ;
5020 -- Error if incorrect enclosing characters:
5021 if ( i == nil ) then
5022     print ("lwarpmk: ===")
5023     print ("lwarpmk: " ..  linenum .. " : " .. line ) ;
5024     print ("lwarpmk: Incorrect entry in " .. conffile ..".\n" ) ;
5025     print ("lwarpmk: ===")
5026 --    printconf () ;
5027     os.exit(1) ;
5028 end -- nil
5029 if ( cvarname == "confversion" ) then
5030     confversion = cvalue
5031 elseif ( cvarname == "opsystem" ) then
5032     -- Verify choice of opsystem:
```

```
5033     if ( (cvalue == "Unix") or (cvalue == "Windows") ) then
5034         opsystem = cvalue
5035     else
5036         cvalueerror ( line, linenum , cvalue )
5037     end
5038 elseif ( cvarname == "sourcename" ) then sourcename = cvalue
5039 elseif ( cvarname == "homehtmlfilename" ) then homehtmlfilename = cvalue
5040 elseif ( cvarname == "htmlfilename" ) then htmlfilename = cvalue
5041 elseif ( cvarname == "imagesdirectory" ) then imagesdirectory = cvalue
5042 elseif ( cvarname == "imagesname" ) then imagesname = cvalue
5043 elseif ( cvarname == "latexmk" ) then latexmk = cvalue
5044 elseif ( cvarname == "printlatexcmd" ) then printlatexcmd = cvalue
5045 elseif ( cvarname == "HTMLlatexcmd" ) then HTMLlatexcmd = cvalue
5046 elseif ( cvarname == "printindexcmd" ) then printindexcmd = cvalue
5047 elseif ( cvarname == "HTMLindexcmd" ) then HTMLindexcmd = cvalue
5048 elseif ( cvarname == "latexmkindexcmd" ) then latexmkindexcmd = cvalue
5049 elseif ( cvarname == "glossarycmd" ) then glossarycmd = cvalue
5050 elseif ( cvarname == "pdftotextenc" ) then pdftotextenc = cvalue
5051 else
5052     print ("lwarpmk: ===")
5053     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
5054     print (
5055         "lwarpmk: Incorrect variable name \"" .. cvarname .. "\" in " ..
5056         conffile ..".\n"
5057     ) ;
5058     print ("lwarpmk: ===")
5059 --    printconf () ;
5060 os.exit(1) ;
5061 end -- cvarname
5062 end -- do scan lines
5063 io.close(cfile)
5064 end -- file exists
5065 -- Error if sourcename is "lwarp".
5066 -- This could happen if a local copy of lwarp has recently been recompiled.
5067 if sourcename=="lwarp" then
5068     print ("lwarpmk: ===")
5069   print ("lwarpmk: lwarp.sty has recently been recompiled in this directory,")
5070   print ("lwarpmk: and \"lwarpmk.conf\" is no longer set for your own project.")
5071     print ("lwarpmk:   (Perhaps you are not in your project's directory?)")
5072     print ("lwarpmk: In your project directory, recompile your project")
5073     print ("lwarpmk: using pdf/lua/xelatex <projectname>.")
5074   print ("lwarpmk: After a recompile, \"lwarpmk.conf\" will be set for your project,")
5075     print ("lwarpmk: and you may again use lwarpmk.")
5076     print ("lwarpmk: ===")
5077     os.exit(1)
5078 end -- sourcename of "lwarp"
5079 -- Select some operating-system commands:
5080 if opsystem=="Unix" then  -- For Unix / Linux / Mac OS:
5081     rmname = "rm"
5082     mvname = "mv"
5083     cpname = "cp"
5084     touchnamepre = "touch"
5085     touchnamepost = ""
5086     newtouchname = "touch"
5087     dirslash = "/"
5088     opquote= "\'"
5089     cmdgroupopenname = " ( "
5090     cmdgroupclosename = " ) "
5091     seqname = " && "
5092     bgname = " &"
```

```
elseif opsystem=="Windows" then -- For Windows
    rmname = "DEL"
    mvname = "MOVE"
    cpname = "COPY"
    touchnamepre = "COPY /b"
    touchnamepost = "+,,"
    newtouchname = "echo empty >"
    dirslash = "\\"
    opquote= "\""
    cmdgroupopenname = ""
    cmdgroupclosename = ""
    seqname = " & "
    bgname = ""
else
    print ("lwarpmk: ===")
    print ("lwarpmk: Select Unix or Windows for opsystem." )
    print ("lwarpmk: ===")
    os.exit(1)
end --- for Windows
-- Warning if the operating system does not appear to be correct,
-- in case files were transferred to another system.
if ( (package.config:sub(1,1)) ~= dirslash ) then
    print ("lwarpmk: ===")
   print ("lwarpmk: It appears that lwarpmk.conf is for a different operating system." )
    printhowtorecompile ()
    print ("lwarpmk: ===")
    os.exit(1)
end
-- Error if the configuration file's version is not current:
if ( confversion ~= requiredconfversion ) then
    print ("lwarpmk: ===")
    printhowtorecompile ()
    print ("lwarpmk: ===")
    os.exit(1)
end
end -- loadconf


function executecheckerror ( executecommands , errormessage )
--
-- Execute an operating system call,
-- and maybe exit with an error message.
--
local err
err = os.execute ( executecommands )
if ( err ~= 0 ) then
    print ("lwarpmk: ===")
    print ("lwarpmk: " .. errormessage )
    print ("lwarpmk: ===")
    os.exit(1)
end
end -- executecheckerror


function refreshdate ()
os.execute(touchnamepre .. " " .. sourcename .. ".tex " .. touchnamepost)
end
```

```
function reruntoget (filesource)
--
-- Scan the LaTeX log file for the phrase "Rerun to get",
-- indicating that the file should be compiled again.
-- Return true if found.
--
local fsource = io.open(filesource)
for line in fsource:lines() do
if ( string.find(line,"Rerun to get") ~= nil ) then
    io.close(fsource)
    return true
end -- if
end -- do
io.close(fsource)
return false
end


function onetime (latexcmd, fsuffix)
--
-- Compile one time, return true if should compile again.
-- fsuffix is "" for print, "_html" for HTML output.
--
print("lwarpmk: Compiling with: " .. latexcmd)
executecheckerror (
    latexcmd ,
    "Compile error."
)
return (reruntoget(sourcename .. fsuffix .. ".log") ) ;
end


function manytimes (latexcmd, fsuffix)
--
-- Compile up to five times.
-- fsuffix is "" for print, "_html" for HTML output
--
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
end end end end end
end


function verifyfileexists (filename)
--
-- Exit if the given file does not exist.
--
if (lfs.attributes ( filename , "modification" ) == nil ) then
    print ("lwarpmk: ===")
    print ("lwarpmk: " .. filename .. " not found." ) ;
    print ("lwarpmk: ===")
    os.exit (1) ;
end
end


```

```
5213
5214 function pdftohtml ()
5215 --
5216 -- Convert <project>_html.pdf into HTML files:
5217 --
5218 -- Convert to text:
5219 print ("lwarpmk: Converting " .. sourcename
5220     .."_html.pdf to " .. sourcename .. "_html.html")
5221 err = os.execute("pdftotext  -enc " .. pdftotextenc .. "  -nopgbrk  -layout "
5222     .. sourcename .. "_html.pdf " .. sourcename .. "_html.html")
5223 if ( err ~= 0 ) then
5224     print ("lwarpmk: ===")
5225     print ("lwarpmk: Ensure that the Poppler utilities are installed." )
5226   print ("lwarpmk: See the Lwarp manual: ‘Installing additional utilities’." )
5227     print ("lwarpmk: ===")
5228     os.exit(1)
5229 end
5230 -- Split the result into individual HTML files:
5231 splitfile (homehtmlfilename .. ".html" , sourcename .. "_html.html")
5232 end
5233
5234
5235 function removeaux ()
5236 --
5237 -- Remove auxiliary files:
5238 -- All .aux files are removed since there may be many bbl*.aux files.
5239 -- Also removes sourcename_html.pdf, sourcename_html.html,
5240 -- and sourcename_html.sidetoc, plus comment_*.cut.
5241 --
5242 os.execute ( rmname .. " *.aux " ..
5243     sourcename ..".toc " .. sourcename .. "_html.toc " ..
5244     sourcename ..".lof " .. sourcename .. "_html.lof " ..
5245     sourcename ..".lot " .. sourcename .. "_html.lot " ..
5246     sourcename ..".bbl " .. sourcename .. "_html.bbl " ..
5247     " *.idx " ..
5248     " *.ind " ..
5249     sourcename ..".ps " .. sourcename .."_html.ps " ..
5250     sourcename ..".log " .. sourcename .. "_html.log " ..
5251     sourcename ..".gl* " .. sourcename .. "_html.gl* " ..
5252     sourcename .. "_html.pdf " ..
5253     sourcename .. "_html.html " ..
5254     sourcename .. "_html.sidetoc " ..
5255     " *_html_inc.* " ..
5256     " comment_*.cut"
5257     )
5258 end
5259
5260 function checkhtmlpdfexists ()
5261 --
5262 -- Error if the HTML document does not exist.
5263 -- The latexmages are drawn from the HTML PDF version of the document,
5264 -- so "lwarpmk html" must be done before "lwarpmk limages".
5265 --
5266 local htmlpdffile = io.open(sourcename .. "_html.pdf", "r")
5267 if ( htmlpdffile == nil ) then
5268     print ("")
5269     print ("lwarpmk: ===")
5270     print ("lwarpmk: The HTML version of the document does not exist.")
5271     print ("lwarpmk: Enter \"lwarpmk html\" to compile the HTML version.")
5272     print ("lwarpmk: ===")
```

```
5273     os.exit(1)
5274 end
5275 io.close (htmlpdffile)
5276 end -- checkhtmlpdfexists
5277
5278
5279 function warnlimages ()
5280 --
5281 -- Warning of a missing <sourcename>-images.txt file:
5282     print ("lwarpmk: ===")
5283     print ("lwarpmk: \"" .. sourcename .. "-images.txt\" does not exist.")
5284    print ("lwarpmk: Your project does not use SVG math or other lateximages,")
5285     print ("lwarpmk: or the file has been deleted somehow.")
5286     print ("lwarpmk: Use \"lwarpmk html1\" to recompile your project")
5287     print ("lwarpmk: and recreate \"" .. sourcename .. "-images.txt\".")
5288    print ("lwarpmk: If your project does not use SVG math or other lateximages,")
5289    print ("lwarpmk: then \"" .. sourcename .. "-images.txt\" will never exist, and")
5290     print ("lwarpmk: \"lwarpmk limages\" will not be necessary.")
5291     print ("lwarpmk: ===")
5292 end -- warnlimages
5293
5294
5295 function warnlimagesrecompile ()
5296 -- Warning if must recompile before creating limages:
5297     print ("")
5298     print ("lwarpmk: ===")
5299     print ("lwarpmk: Cross-references are not yet correct.")
5300    print ("lwarpmk: The document must be recompiled before creating the lateximages.")
5301    print ("lwarpmk: Enter \"lwarpmk html1\" again, then try \"lwarpmk limages\" again.")
5302     print ("lwarpmk: ===")
5303 end --warnlimagesrecompile
5304
5305
5306 function checklimages ()
5307 --
5308 -- Check <sourcename>.txt to see if need to recompile first.
5309 -- If any entry has a page number of zero, then there were incorrect images.
5310 --
5311 print ("lwarpmk: Checking for a valid " .. sourcename .. "-images.txt file.")
5312 local limagesfile = io.open(sourcename .. "-images.txt", "r")
5313 if ( limagesfile == nil ) then
5314     warnlimages ()
5315     os.exit(1)
5316 end
5317 -- Track warning to recompile if find a page 0
5318 local pagezerowarning = false
5319 -- Scan <sourcename>.txt
5320 for line in limagesfile:lines() do
5321     -- lwimgpage is the page number in the PDF which has the image
5322     -- lwimghash is true if this filename is a hash
5323     -- lwimgname is the lateximage filename root to assign for the image
5324    i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
5325     -- For each entry:
5326     if ( (i~=nil) ) then
5327         -- If the page number is 0, image references are incorrect
5328         --  and must recompile the soure document:
5329         if ( lwimgpage == "0" ) then
5330             pagezerowarning = true
5331         end
5332     end -- if i~=nil
```

```
end -- do
-- The last line should be |end|end|end|.
-- If not, the compile must have aborted, and the images are incomplete.
if ( lwimgpage ~= "end" ) then
    warnlimagesrecompile()
    os.exit(1) ;
end
if ( pagezerowarning ) then
    warnlimagesrecompile()
    os.exit(1) ;
end -- pagezerowarning
end -- checklimages


function createuniximage ( lwimgfullname )
--
-- Create one lateximage for Unix / Linux / Mac OS.
--
executecheckerror (
    cmdgroupopenname ..
    "pdfseparate -f " .. lwimgpage .. " -l " .. lwimgpage .. " " ..
        sourcename .."_html.pdf " ..
        imagesdirectory .. dirslash .."lateximagetemp-%d" .. ".pdf" ..
        seqname ..
    -- Crop the image:
   "pdfcrop --hires --margins \"0 1 0 0\" " .. imagesdirectory .. dirslash .. "lateximagetemp-" ..
        lwimgpage .. ".pdf " ..
        imagesdirectory .. dirslash .. lwimgname .. ".pdf" ..
        seqname ..
    -- Convert the image to svg:
   "pdftocairo -svg -noshrink " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf " ..
        imagesdirectory .. dirslash .. lwimgname ..".svg" ..
        seqname ..
    -- Remove the temporary files:
   rmname .. " " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf" .. seqname ..
   rmname .. " " .. imagesdirectory .. dirslash .. "lateximagetemp-" .. lwimgpage .. ".pdf" ..
    cmdgroupclosename .. " >/dev/null " .. bgname
    ,
    "File error trying to convert " .. lwimgfullname
)
-- Every 32 images, wait for completion at below normal priority,
--  allowing other image tasks to catch up.
numimageprocesses = numimageprocesses + 1
if ( numimageprocesses > 32 ) then
    numimageprocesses = 0
    print ( "lwarpmk: waiting" )
    executecheckerror ( "wait" , "File error trying to wait.")
end
end -- createuniximage


function createwindowsimage ( lwimgfullname )
--
-- Create one lateximage for Windows.
--
-- Every 32 images, wait for completion at below normal priority,
--  allowing other image tasks to catch up.
numimageprocesses = numimageprocesses + 1
if ( numimageprocesses > 32 ) then
    numimageprocesses = 0
```

```
5393     thiswaitcommand = "/WAIT /BELOWNORMAL"
5394     print ( "lwarpmk: waiting" )
5395 else
5396     thiswaitcommand = ""
5397 end
5398 -- Execute the image generation command
5399 executecheckerror (
5400     "start /B " .. thiswaitcommand .. " \"\" lwarp_one_limage " ..
5401     lwimgpage .. " " ..
5402     lwimghash .. " " ..
5403     lwimgname .. " " ..
5404     sourcename .. " <nul >nul"
5405     ,
5406     "File error trying to create image."
5407 )
5408 end -- createwindowsimage
5409
5410
5411 function createonelateximage ( line )
5412 --
5413 -- Given the next line of <sourcename>.txt, convert a single image.
5414 --
5415 -- lwimgpage is the page number in the PDF which has the image
5416 -- lwimghash is true if this filename is a hash
5417 -- lwimgname is the lateximage filename root to assign for the image
5418 i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
5419 -- For each entry:
5420 if ( (i~=nil) ) then
5421     -- Skip if the page number is 0:
5422     if ( lwimgpage == "0" ) then
5423         pagezerowarning = true
5424     -- Skip if the page number is "end":
5425     else if ( lwimgpage == "end" ) then
5426     else
5427         -- Skip is this image is hashed and already exists:
5428       local lwimgfullname = imagesdirectory .. dirslash .. lwimgname .. ".svg"
5429         if (
5430             (lwimghash ~= "true") or
5431             (lfs.attributes(lwimgfullname,"mode")==nil) -- file not exists
5432         )
5433         then -- not hashed or not exists:
5434             -- Print the name of the file being generated:
5435             print ( "lwarpmk: " .. lwimgname )
5436          -- Touch/create the dest so that only once instance tries to build it:
5437             executecheckerror (
5438                 newtouchname .. " " .. lwimgfullname ,
5439                 "File error trying to touch " .. lwimgfullname
5440             )
5441             -- Separate out the image into its own single-page pdf:
5442             if opsystem=="Unix" then
5443                 createunixiimage (lwimgfullname)
5444             elseif opsystem=="Windows" then
5445                 createwindowsimage (lwimgfullname)
5446             end
5447         end -- not hashed or not exists
5448     end -- not page "end"
5449     end -- not page 0
5450 end -- not nil
5451 end -- createonelateximage
5452
```

```
5453
5454 function createlateximages ()
5455 --
5456 -- Create lateximages based on <sourcename>-images.txt:
5457 --
5458 -- See if the document must be recompiled first:
5459 checklimages ()
5460 -- See if the HTML version exists:
5461 checkhtmlpdfexists ()
5462 -- Attempt to create the lateximages:
5463 print ("lwarpmk: Creating lateximages.")
5464 local limagesfile = io.open(sourcename .. "-images.txt", "r")
5465 if ( limagesfile == nil ) then
5466     warnlimages ()
5467     os.exit(1)
5468 end
5469 -- Create the lateximages directory, ignore error if already exists
5470 err = os.execute("mkdir " .. imagesdirectory)
5471 -- For Windows, create lwarp_one_limage.cmd from lwarp_one_limage.txt:
5472 if opsystem=="Windows" then
5473     executecheckerror (
5474         cpname .. " lwarp_one_limage.txt lwarp_one_limage.cmd" ,
5475       "File error trying to copy lwarp_one_limage.txt to lwarp_one_limage.cmd"
5476     )
5477 end -- create lwarp_one_limage.cmd
5478 -- Track the number of parallel processes
5479 numimageprocesses = 0
5480 -- Track warning to recompile if find a page 0
5481 pagezerowarning = false
5482 -- Scan <sourcename>.txt
5483 for line in limagesfile:lines() do
5484     createonelateximage ( line )
5485 end -- do
5486 io.close(limagesfile)
5487 print ( "lwarpmk limages: ===")
5488 print ( "lwarpmk limages: Wait a moment for the images to complete" )
5489 print ( "lwarpmk limages:   before reloading the page." )
5490 print ( "lwarpmk limages: ===")
5491 print ( "lwarpmk limages: Done." )
5492 if ( pagezerowarning == true ) then
5493     print ( "lwarpmk limages: WARNING: Images will be incorrect." )
5494     print ( "lwarpmk limages:   Enter \"lwarpmk cleanlimages\", then" )
5495     print ( "lwarpmk limages:   recompile the document one more time, then" )
5496     print ( "lwarpmk limages:   repeat \"lwarpmk images\" again." )
5497 end -- pagezerowarning
5498 end -- function
5499
5500
5501 function convertepstopdf ()
5502 --
5503 -- Converts EPS files to PDF files.
5504 -- The filenames are arg[argindex] and up.
5505 -- arg[1] is the command "epstopdf".
5506 --
5507 ignoreconf ()
5508 for i = argindex , #arg do
5509     if (lfs.attributes(arg[i],"mode")==nil) then
5510         print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
5511     else
5512         print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
```

```
5513         thispath, thisfilename, thisextension = splitfilename(arg[i])
5514         if ( thispath == nil ) then
5515             os.execute ( "epstopdf " .. arg[i] )
5516         else
5517             os.execute (
5518                 "epstopdf " ..
5519                 thispath .. thisfilename .. "." .. thisextension .. " " ..
5520                 thispath .. thisfilename .. ".pdf"
5521             )
5522         end
5523     end -- if
5524 end -- do
5525 end --function
5526
5527
5528 function convertpdftosvg ()
5529 --
5530 -- Converts PDF files to SVG files.
5531 -- The filenames are arg[argindex] and up.
5532 -- arg[1] is the command "pdftosvg".
5533 --
5534 ignoreconf ()
5535 for i = argindex , #arg do
5536     if (lfs.attributes(arg[i],"mode")==nil) then
5537         print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
5538     else
5539         print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
5540         thispath, thisfilename, thisextension = splitfilename(arg[i])
5541         if ( thispath == nil ) then
5542             os.execute ( "pdftocairo -svg " .. arg[i] )
5543         else
5544             os.execute (
5545                 "pdftocairo -svg " ..
5546                 thispath .. thisfilename .. "." .. thisextension .. " " ..
5547                 thispath .. thisfilename .. ".svg"
5548             )
5549         end
5550     end -- if
5551 end -- do
5552 end --function
5553
5554
5555 -- Force an update and conclude processing:
5556 function updateanddone ()
5557 print ("lwarpmk: Forcing an update of " .. sourcename ..".tex.")
5558 refreshdate ()
5559 print ("lwarpmk: " .. sourcename ..".tex is ready to be recompiled.")
5560 print ("lwarpmk: Done.")
5561 end -- function
5562
5563
5564 -- Start of the main code: --
5565
5566
5567 -- lwarpmk --version :
5568
5569 if (arg[1] == "--version") then
5570 print ( "lwarpmk: " .. printversion )
5571
5572 else -- not --version
```

```
5573
5574
5575 -- print intro:
5576
5577 print ("lwarpmk: " .. printversion .. "  Automated make for the LaTeX Lwarp package.")
5578
5579
5580 -- lwarpmk print:
5581
5582 if arg[1] == "print" then
5583 loadconf ()
5584 if ( latexmk == "true" ) then
5585     print ("lwarpmk: Compiling with: " .. printlatexcmd)
5586     executecheckerror (
5587         printlatexcmd ,
5588         "Compile error."
5589     )
5590     print ("lwarpmk: Done.")
5591 else -- not latexmk
5592     verifyfileexists (sourcename .. ".tex") ;
5593     -- See if up to date:
5594     if (
5595        ( lfs.attributes ( sourcename .. ".pdf" , "modification" ) == nil ) or
5596         (
5597             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
5598             lfs.attributes ( sourcename .. ".pdf" , "modification" )
5599         )
5600     ) then
5601         -- Recompile if not yet up to date:
5602         manytimes(printlatexcmd, "")
5603         print ("lwarpmk: Done.") ;
5604     else
5605         print ("lwarpmk: " .. sourcename .. ".pdf is up to date.") ;
5606     end
5607 end -- not latexmk
5608
5609
5610 -- lwarpmk print1:
5611
5612 elseif arg[1] == "print1" then
5613     loadconf ()
5614     verifyfileexists (sourcename .. ".tex") ;
5615     onetime(printlatexcmd, "")
5616     print ("lwarpmk: Done.") ;
5617
5618
5619 -- lwarpmk printindex:
5620 -- Compile the index then touch the source
5621 -- to trigger a recompile of the document:
5622
5623 elseif arg[1] == "printindex" then
5624 loadconf ()
5625 os.execute ( printindexcmd )
5626 print ("lwarpmk: -------")
5627 updateanddone ()
5628
5629
5630 -- lwarpmk printglossary:
5631 -- Compile the glossary then touch the source
5632 -- to trigger a recompile of the document:
```

```
5633
5634 elseif arg[1] == "printglossary" then
5635 loadconf ()
5636 print ("lwarpmk: Processing the glossary.")
5637
5638 os.execute(glossarycmd .. " " .. sourcename)
5639 updateanddone ()
5640
5641
5642 -- lwarpmk html:
5643
5644 elseif arg[1] == "html" then
5645 loadconf ()
5646 if ( latexmk == "true" ) then
5647     print ("lwarpmk: Compiling with: " .. HTMLlatexcmd)
5648     executecheckerror (
5649         HTMLlatexcmd ,
5650         "Compile error."
5651     )
5652     pdftohtml ()
5653     print ("lwarpmk: Done.")
5654 else -- not latexmk
5655     verifyfileexists ( sourcename .. ".tex" ) ;
5656     -- See if exists and is up to date:
5657     if (
5658       ( lfs.attributes ( homehtmlfilename .. ".html" , "modification" ) == nil ) or
5659         (
5660             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
5661             lfs.attributes ( homehtmlfilename .. ".html" , "modification" )
5662         )
5663     ) then
5664         -- Recompile if not yet up to date:
5665         manytimes(HTMLlatexcmd, "_html")
5666         pdftohtml ()
5667         print ("lwarpmk: Done.")
5668     else
5669         print ("lwarpmk: " .. homehtmlfilename .. ".html is up to date.")
5670     end
5671 end -- not latexmk
5672
5673
5674 -- lwarpmk html1:
5675
5676 elseif arg[1] == "html1" then
5677     loadconf ()
5678     verifyfileexists ( sourcename .. ".tex" ) ;
5679     onetime(HTMLlatexcmd, "_html")
5680     pdftohtml ()
5681     print ("lwarpmk: Done.")
5682
5683
5684 -- lwarpmk pdftohtml:
5685 elseif arg[1] == "pdftohtml" then
5686     loadconf ()
5687     pdftohtml ()
5688
5689
5690 -- lwarpmk htmlindex:
5691 -- Compile the index then touch the source
5692 -- to trigger a recompile of the document:
```

```
5693
5694 elseif arg[1] == "htmlindex" then
5695 loadconf ()
5696 os.execute ( HTMLindexcmd )
5697 print ("lwarpmk: -------")
5698 updateanddone ()
5699
5700
5701 -- lwarpmk htmlglossary:
5702 -- Compile the glossary then touch the source
5703 -- to trigger a recompile of the document.
5704 -- The <sourcename>.xdy file is created by the glossaries package.
5705
5706 elseif arg[1] == "htmlglossary" then
5707 loadconf ()
5708 print ("lwarpmk: Processing the glossary.")
5709 os.execute(glossarycmd .. " " .. sourcename .. "_html")
5710 updateanddone ()
5711
5712
5713 -- lwarpmk limages:
5714 -- Scan the <sourcename>.txt file to create lateximages.
5715
5716 elseif arg[1] == "limages" then
5717 loadconf ()
5718 print ("lwarpmk: Processing images.")
5719 createlateximages ()
5720 print ("lwarpmk: Done.")
5721
5722
5723 -- lwarpmk again:
5724 -- Touch the source to trigger a recompile.
5725
5726 elseif arg[1] == "again" then
5727 loadconf ()
5728 updateanddone ()
5729
5730
5731 -- lwarpmk clean:
5732 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
5733
5734 elseif arg[1] == "clean" then
5735 loadconf ()
5736 removeaux ()
5737 print ("lwarpmk: Done.")
5738
5739
5740 -- lwarpmk cleanall
5741 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
5742 --    and also project.pdf, project.dvi, *.html
5743
5744 elseif arg[1] == "cleanall" then
5745 loadconf ()
5746 removeaux ()
5747 os.execute ( rmname .. " " ..
5748     sourcename .. ".pdf " .. sourcename .. "_html.pdf " ..
5749     sourcename .. ".dvi " .. sourcename .. "_html.dvi " ..
5750     "*.html"
5751     )
5752 print ("lwarpmk: Done.")
```

```
5753
5754
5755 -- lwarpmk cleanlimages
5756 -- Remove images from the imagesdirectory.
5757
5758 elseif arg[1] == "cleanlimages" then
5759 loadconf ()
5760 os.execute ( rmname .. " " .. imagesdirectory .. dirslash .. "*" )
5761 print ("lwarpmk: Done.")
5762
5763 -- lwarpmk epstopdf <list of file names>
5764 -- Convert EPS files to PDF using epstopdf
5765 elseif arg[1] == "epstopdf" then
5766 convertepstopdf ()
5767 print ("lwarpmk: Done.")
5768
5769
5770 -- lwarpmk pdftosvg <list of file names>
5771 -- Convert PDF files to SVG using pdftocairo
5772 elseif arg[1] == "pdftosvg" then
5773 convertpdftosvg ()
5774 print ("lwarpmk: Done.")
5775
5776
5777 -- lwarpmk with no argument :
5778
5779 elseif (arg[1] == nil) then
5780 printhelp ()
5781
5782
5783 -- lwarpmk -v:
5784
5785 elseif (arg[1] == "-v" ) then
5786 -- The version number has already been printed
5787 -- by the lwarpmk intro.
5788
5789 -- lwarpmk -h or lwarpmk --help :
5790
5791 elseif (arg[1] == "-h" ) or (arg[1] == "--help") then
5792 printusage ()
5793
5794
5795 -- Unknown command:
5796
5797 else
5798 printhelp ()
5799 print ("\nlwarpmk: ****** Unknown command \""..arg[1].."\". ******\n")
5800 end
5801
5802 end -- not --version
5803 \end{filecontents*}
5804 % \end{Verbatim}% for syntax highlighting

5805 \end{LWRcreatelwarpmk}
```

41 Stacks

for HTML output: 5806 \begin{warpHTML}

Stacks are used to remember how to close sections and list items. Before a new section is started, previously nested sections and items must be closed out (un-nested) in proper order. Note that starting a new section may close several levels of previously nested items at the same time. For example, starting a new \section would close any currently open subsection, subsubsection, and paragraph. General environments are not nested on the stack since they have their own close mechanism. List environments are nested, and items inside those environments are nested one level deeper still. List environments may be nested inside other list environments, and list items are nested inside list environments as well. Thus, the stack may have items which are not necessarily in order, since a description may contain an enumerate, for example. Depths to be recorded in \LWR@closedepthone, etc.

41.1 Assigning depths

initial depths for empty stack entries:

```
5807 \newcommand*{\LWR@depthnone}{-5}
```

All sectioning depths are deeper than LWR@depthfinished:

```
5808 \newcommand*{\LWR@depthfinished}{-4}
5809 \newcommand*{\LWR@depthbook}{-2}
5810 \newcommand*{\LWR@depthpart}{-1}
5811 \newcommand*{\LWR@depthchapter}{0}
5812 \newcommand*{\LWR@depthsection}{1}
5813 \newcommand*{\LWR@depthsubsection}{2}
5814 \newcommand*{\LWR@depthsubsubsection}{3}
5815 \newcommand*{\LWR@depthparagraph}{4}
5816 \newcommand*{\LWR@depthsubparagraph}{5}
```

Used by \itemize, \enumerate, \description:

```
5817 \newcommand*{\LWR@depthlist}{6}
```

Used by \item:

```
5818 \newcommand*{\LWR@depthlistitem}{7}
5819 \let\LWR@depthdescitem\LWR@depthlistitem
```

41.2 Closing actions

A stack to record the action to take to close each nesting level: Add more levels of stack if necessary for a very deeply nested document, adding to \pushclose and \popclose as well.

```
5820 \newcommand*{\LWR@closeone}{}% top of the stack
5821 \newcommand*{\LWR@closetwo}{}
5822 \newcommand*{\LWR@closethree}{}
5823 \newcommand*{\LWR@closefour}{}
```

```
\newcommand*{\LWR@closefive}{}
\newcommand*{\LWR@closesix}{}
\newcommand*{\LWR@closeseven}{}
\newcommand*{\LWR@closeeight}{}
\newcommand*{\LWR@closenine}{}
\newcommand*{\LWR@closeten}{}
\newcommand*{\LWR@closeeleven}{}
\newcommand*{\LWR@closetwelve}{}
\newcommand*{\LWR@closethirteen}{}
\newcommand*{\LWR@closefourteen}{}
\newcommand*{\LWR@closefifteen}{}
\newcommand*{\LWR@closesixteen}{}
\newcommand*{\LWR@closeseventeen}{}
\newcommand*{\LWR@closeeighteen}{}
\newcommand*{\LWR@closenineteen}{}
```

41.3 Closing depths

A stack to record the depth of each level:

⚠ Note that nested LaTeX structures may push depths which are non-sequential.

Ex:

```
\begin{itemize}
   \item{A}
   \begin{description}
      \item{B}
   \end{description}
\end{itemize}
```

```
\newcommand*{\LWR@closedepthone}{\LWR@depthnone}% top of the stack
\newcommand*{\LWR@closedepthtwo}{\LWR@depthnone}
\newcommand*{\LWR@closedepththree}{\LWR@depthnone}
\newcommand*{\LWR@closedepthfour}{\LWR@depthnone}
\newcommand*{\LWR@closedepthfive}{\LWR@depthnone}
\newcommand*{\LWR@closedepthsix}{\LWR@depthnone}
\newcommand*{\LWR@closedepthseven}{\LWR@depthnone}
\newcommand*{\LWR@closedeptheight}{\LWR@depthnone}
\newcommand*{\LWR@closedepthnine}{\LWR@depthnone}
\newcommand*{\LWR@closedepthten}{\LWR@depthnone}
\newcommand*{\LWR@closedeptheleven}{\LWR@depthnone}
\newcommand*{\LWR@closedepthtwelve}{\LWR@depthnone}
\newcommand*{\LWR@closedepththirteen}{\LWR@depthnone}
\newcommand*{\LWR@closedepthfourteen}{\LWR@depthnone}
\newcommand*{\LWR@closedepthfifteen}{\LWR@depthnone}
\newcommand*{\LWR@closedepthsixteen}{\LWR@depthnone}
\newcommand*{\LWR@closedepthseventeen}{\LWR@depthnone}
\newcommand*{\LWR@closedeptheighteen}{\LWR@depthnone}
\newcommand*{\LWR@closedepthnineteen}{\LWR@depthnone}
```

41.4 Pushing and popping the stack

`\LWR@pushclose {`⟨*sectiontype*⟩`}`

Pushes one return action and its LaTeX depth onto the stacks.

```
\NewDocumentCommand{\LWR@pushclose}{m}
{%
\global\let\LWR@closenineteen\LWR@closeeighteen%
\global\let\LWR@closeeighteen\LWR@closeseventeen%
\global\let\LWR@closeseventeen\LWR@closesixteen%
\global\let\LWR@closesixteen\LWR@closefifteen%
\global\let\LWR@closefifteen\LWR@closefourteen%
\global\let\LWR@closefourteen\LWR@closethirteen%
\global\let\LWR@closethirteen\LWR@closetwelve%
\global\let\LWR@closetwelve\LWR@closeeleven%
\global\let\LWR@closeeleven\LWR@closeten%
\global\let\LWR@closeten\LWR@closenine%
\global\let\LWR@closenine\LWR@closeeight%
\global\let\LWR@closeeight\LWR@closeseven%
\global\let\LWR@closeseven\LWR@closesix%
\global\let\LWR@closesix\LWR@closefive%
\global\let\LWR@closefive\LWR@closefour%
\global\let\LWR@closefour\LWR@closethree%
\global\let\LWR@closethree\LWR@closetwo%
\global\let\LWR@closetwo\LWR@closeone%
\global\csletcs{LWR@closeone}{LWR@printclose#1}%
\global\let\LWR@closedepthnineteen\LWR@closedeptheighteen%
\global\let\LWR@closedeptheighteen\LWR@closedepthseventeen%
\global\let\LWR@closedepthseventeen\LWR@closedepthsixteen%
\global\let\LWR@closedepthsixteen\LWR@closedepthfifteen%
\global\let\LWR@closedepthfifteen\LWR@closedepthfourteen%
\global\let\LWR@closedepthfourteen\LWR@closedepththirteen%
\global\let\LWR@closedepththirteen\LWR@closedepthtwelve%
\global\let\LWR@closedepthtwelve\LWR@closedepthteleven%
\global\let\LWR@closedepthteleven\LWR@closedepththen%
\global\let\LWR@closedepththen\LWR@closedepthnine%
\global\let\LWR@closedepthnine\LWR@closedeptheight%
\global\let\LWR@closedeptheight\LWR@closedepthseven%
\global\let\LWR@closedepthseven\LWR@closedepthsix%
\global\let\LWR@closedepthsix\LWR@closedepthfive%
\global\let\LWR@closedepthfive\LWR@closedepthfour%
\global\let\LWR@closedepthfour\LWR@closedepththree%
\global\let\LWR@closedepththree\LWR@closedepthtwo%
\global\let\LWR@closedepthtwo\LWR@closedepthone%
\global\csletcs{LWR@closedepthone}{LWR@depth#1}%
```

Error if the deepest depth is no longer \LWR@depthnone, which means that it somehow has been nested too deeply, or things are not being unnested correctly.

```
\ifdefstring{\LWR@closedepthnineteen}{\LWR@depthnone}%
    {}%
    {%
        \PackageError{lwarp}%
            {The document is nested too deeply for Lwarp}%
            {PLEASE inform the Lwarp maintainer!}%
    }%
}
```

\LWR@popclose Pops one action and its depth off the stacks.

```
\newcommand*{\LWR@popclose}
{%
```

```
\global\let\LWR@closeone\LWR@closetwo%
\global\let\LWR@closetwo\LWR@closethree%
\global\let\LWR@closethree\LWR@closefour%
\global\let\LWR@closefour\LWR@closefive%
\global\let\LWR@closefive\LWR@closesix%
\global\let\LWR@closesix\LWR@closeseven%
\global\let\LWR@closeseven\LWR@closeeight%
\global\let\LWR@closeeight\LWR@closenine%
\global\let\LWR@closenine\LWR@closeten%
\global\let\LWR@closeten\LWR@closeeleven%
\global\let\LWR@closeeleven\LWR@closetwelve%
\global\let\LWR@closetwelve\LWR@closethirteen%
\global\let\LWR@closethirteen\LWR@closefourteen%
\global\let\LWR@closefourteen\LWR@closefifteen%
\global\let\LWR@closefifteen\LWR@closesixteen%
\global\let\LWR@closesixteen\LWR@closeseventeen%
\global\let\LWR@closeseventeen\LWR@closeeighteen%
\global\let\LWR@closeeighteen\LWR@closenineteen%
\global\let\LWR@closedepthone\LWR@closedepthtwo%
\global\let\LWR@closedepthtwo\LWR@closedepththree%
\global\let\LWR@closedepththree\LWR@closedepthfour%
\global\let\LWR@closedepthfour\LWR@closedepthfive%
\global\let\LWR@closedepthfive\LWR@closedepthsix%
\global\let\LWR@closedepthsix\LWR@closedepthseven%
\global\let\LWR@closedepthseven\LWR@closedeptheight%
\global\let\LWR@closedeptheight\LWR@closedepthnine%
\global\let\LWR@closedepthnine\LWR@closedepthten%
\global\let\LWR@closedepthten\LWR@closedeptheleven%
\global\let\LWR@closedeptheleven\LWR@closedepthtwelve%
\global\let\LWR@closedepthtwelve\LWR@closedepththirteen%
\global\let\LWR@closedepththirteen\LWR@closedepthfourteen%
\global\let\LWR@closedepthfourteen\LWR@closedepthfifteen%
\global\let\LWR@closedepthfifteen\LWR@closedepthsixteen%
\global\let\LWR@closedepthsixteen\LWR@closedepthseventeen%
\global\let\LWR@closedepthseventeen\LWR@closedeptheighteen%
\global\let\LWR@closedeptheighteen\LWR@closedepthnineteen%
}

\end{warpHTML}
```

42 Data arrays

These macros are similar to the arrayjobx package, except that `\LWR@setexparray`'s argument is expanded only once when assigned.

`name` has no backslash, `index` can be a number or a text name, and an empty `value` must be `\relax` instead of empty.

To assign an empty value:

```
\LWR@setexparray{name}{index}{}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@setexparray` {⟨*name*⟩} {⟨*index*⟩} {⟨*contents*⟩}

```
\newbool{LWR@setexparray@doingparhooks}

\NewDocumentCommand{\LWR@setexparray}{m m m}{%
```

Temporarily disable paragraph handling during the assignment. This is not done in a group with global assignments because a table may be nested.

```
    \let\ifLWR@setexparray@doingparhooks\ifLWR@doingparhooks%
    \setbool{LWR@doingparhooks}{false}%
    \let\LWR@setexparray@par\par%
    \let\par\relax%
```

The name of the control sequence is the given name with the index appended.

```
    \xdef\LWR@thisexparrayname{#1#2}%
```

Locally assign the value to the control sequence:

```
    \ifstrempty{#3}%
        {\csdef{\LWR@thisexparrayname}{}}%
        {\csedef{\LWR@thisexparrayname}{#3}}%
```

Restore the paragraph handling:

```
    \let\ifLWR@doingparhooks\ifLWR@setexparray@doingparhooks%
    \let\par\LWR@setexparray@par%
}
```

`\LWR@getexparray` {⟨*name*⟩} {⟨*index*⟩}

```
\newcommand*{\LWR@getexparray}[2]{%
    \@nameuse{#1#2}%
}

\end{warpHTML}
```

43 Localizing catcodes

for HTML & PRINT:

```
\begin{warpall}
```

⚠ Misplaced alignment tab character &

Place `\StartDefiningTabulars` and `\StopDefiningTabulars` before and after defining macros or environments which include the tabular & character in their definitions.

The catcode of & must be changed before the definitions begin, and must be restored afterwards. Doing so avoids the error

```
Misplaced alignment tab character &.
```

`\StartDefiningTabulars` Place before defining something with & in it.

```
\newcommand{\StartDefiningTabulars}{%
    \LWR@traceinfo{StartDefiningTabulars}%
    \warpHTMLonly{\catcode`\&=\active}%
}
```

`\StopDefiningTabulars` Place after defining something with `&` in it.

```
\newcommand{\StopDefiningTabulars}{%
    \LWR@traceinfo{StopDefiningTabulars}%
    \warpHTMLonly{\catcode`\&=4}%
}
```

`LWR@mathmacro` (*bool*) True if currently defining math macros. Used to disable SVG math hashing and MathJax math contents while defining a macro using inline math. Begin a macro, it is not guaranteed that the contents are static, and so the image must be unique. The contents also almost certainly will not be parsed correctly by MathJax.

```
\newbool{LWR@mathmacro}
\boolfalse{LWR@mathmacro}
```

`\StartDefiningMath` Place before defining something with `$` in it.

```
\newcommand{\StartDefiningMath}{%
    \LWR@traceinfo{StartDefiningMath}%
    \warpHTMLonly{\catcode`\$=\active}%
}
```

`\StopDefiningMath` Place after defining something with `$` in it.

```
\newcommand{\StopDefiningMath}{%
    \LWR@traceinfo{StopDefiningMath}%
    \warpHTMLonly{\catcode`\$=3}% math shift
}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

A definition for `&` in case it is referred to after `\StartDefiningTabulars` but outside a `tabular`.

```
\StartDefiningTabulars
\protected\gdef&{%
    \PackageWarning{lwarp}{%
        An ampersand is being used inside a tabular\MessageBreak
    }%
}%
\StopDefiningTabulars
```

```
\end{warpHTML}
```

44 Localizing dynamic math

Inline SVG math usually uses a hash of its contents to generate `latexi`mages which are reusable for multiple instances with the same contents. If the contents may change for each use, such as depending on the current value of a counter, then `\inlinemathother` must be used before the inline math expression, and `\inlinemathnormal` must be used after.

For MathJax, the inline math expression is usually printed for MathJax to interpret. When marked as dynamic math, the following inline math expression will be displayed as an unhashed inline SVG image instead.

For existing code and packages, it may be possible to patch macros after they have been defined, using the xpatch package, which is pre-loaded by lwarp:

```
\xpatchcmd{\macroname}
    {$math expression$}
    {\inlinemathother$math expression$\inlinemathnormal}
    {}
    {\typeout{Error patching macroname.}}
```

for HTML & PRINT:
```
\begin{warpall}
```

`LWR@dynamicmath` (*bool*) Default: `false`

True to mark inline math which is dynamic in nature, thus should not be hashed for reuse.

```
\newbool{LWR@dynamicmath}
\boolfalse{LWR@dynamicmath}
```

`\inlinemathother` Place before using `$ ... $` or `\( ... \)` if the contents of the math are not static, depending on counters or dynamic macros.

```
\newcommand{\inlinemathother}{%
\LWR@traceinfo{inlinemathother}%
\booltrue{LWR@dynamicmath}%
}
```

`\inlinemathnormal` Place after using `$ ... $` or `\( ... \)` with dynamic contents.

```
\newcommand{\inlinemathnormal}{%
\LWR@traceinfo{inlinemathnormal}%
\boolfalse{LWR@dynamicmath}%
}
```

```
\end{warpall}
```

45 HTML entities

for HTML output:
```
\begin{warpHTML}
```

HTML Unicode entities:

```
\let\LWR@origampersand\&
```

`\LWR@fontfortags` {⟨*macro name*⟩} {⟨*argument*⟩}

Forces roman TT font for HTML tags.

```
\newrobustcmd*{\LWR@fontfortags}[2]{%
```

```
    \ifmmode%
        \PackageError{lwarp}%
            {%
                An HTML tag was generated inside math.\MessageBreak
                This should never occur.\MessageBreak
                Something is broken in Lwarp.\MessageBreak
                Enter ‘h’ for details%
            }%
            {(Using #1{#2}.)}%
    \else%
```

Used by ltjtbook, platex, and related.

```
        \ifdef{\romanencoding}%
            {%
                \romanencoding{\encodingdefault}%
            }%
            {%
```

Used by babel:

```
                \ifdef{\latintext}
                    {\latintext}
                    {\fontencoding\encodingdefault}%
            }%
        \LWR@print@normalfont%
        \LWR@origttfamily%
    \fi%
}
```

`\HTMLentity {⟨entitytag⟩}`

`\protect` is in case the tag appears in TOC, LOF, LOT.

```
\newcommand*{\HTMLentity}[1]{%
% \LWR@traceinfo{HTMLentity \detokenize{#1}}%
    \begingroup%
    \LWR@hook@processingtags%
    \LWR@fontfortags{HTMLentity}{\detokenize{#1}}%
    \protect\LWR@origampersand\LWR@isolate{#1};%
    \endgroup%
% \LWR@traceinfo{HTMLentity done}%
}
```

`\HTMLunicode {⟨hex_unicode⟩}`

```
\newcommand*{\HTMLunicode}[1]{\HTMLentity{\LWR@origpound{}x#1}}
```

`\&`

```
\renewrobustcmd*{\&}{\HTMLentity{amp}}
```

`\textless`

```
\let\LWR@origtextless\textless
\renewrobustcmd*{\textless}{\HTMLentity{lt}}
```

\textgreater

```
\let\LWR@origtextgreater\textgreater
\renewrobustcmd*{\textgreater}{\HTMLentity{gt}}
```

```
\end{warpHTML}
```

46 HTML filename generation

The filename of the homepage is set to `\HomeHTMLFilename.html`. The filenames of additional sections start with `\HTMLFilename`, to which is appended a section number or a simplified section name, depending on `FileSectionNames`.

for HTML & PRINT:

```
\begin{warpall}
```

\BaseJobname The `\jobname` of the printed version, even if currently compiling the HTML version. I.e. this is the `\jobname` without `_html` appended. This is used to set `\HomeHTMLFilename` if the user did not provide one.

```
\providecommand*{\BaseJobname}{\jobname}
```

\HTMLFilename The prefix for all generated HTML files other than the home page, defaulting to empty. See section 7.6.1.

```
\providecommand*{\HTMLFilename}{}
```

\HomeHTMLFilename The filename of the home page, defaulting to the `\BaseJobname`. See section 7.6.1.

```
\providecommand*{\HomeHTMLFilename}{\BaseJobname}
```

\SetHTMLFileNumber {⟨*number*⟩}

Sets the file number for the next file to be generated. 0 is the home page. Use just before the next sectioning command, and set it to one less than the desired number of the next section. May be used to generate numbered groups of nodes such as 100+ for one chapter, 200+ for another chapter, etc.

```
\newcommand*{\SetHTMLFileNumber}[1]{%
    \setcounter{LWR@htmlfilenumber}{#1}%
}
```

FileSectionNames (*bool*) Selects how to create HTML file names.

Defaults to use section names in the filenames.

```
\newbool{FileSectionNames}
\booltrue{FileSectionNames}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

Updated each time a new HTML file is begun. Used to provide HTML previous/next web page links.

```
\newcounter{LWR@HTMLpagenum}
\setcounter{LWR@HTMLpagenum}{0}
```

`LWR@htmlseqfilenumber` (*Ctr*) A sequential count of the number of each HTML file as it is being created. Number 0 is the home page. Unlike `\LWR@htmlfilenumber`, this one is known to increment by one for each file. This is used to generate previous /next links for each web page, via labels called `\BaseJobname-autofile-*`, and the last page is also labelled `\BaseJobname-autofile-last`.

```
\newcounter{LWR@htmlseqfilenumber}
\setcounter{LWR@htmlseqfilenumber}{0}
```

`LWR@setseqfilelabel` (*bool*) At each new HTML file, this is false until a sectional unit is used, at which point this is set true and a label is placed. In this way, the previous/next labels will point to a named section.

```
\newbool{LWR@setseqfilelabel}
\setbool{LWR@setseqfilelabel}{false}
```

`LWR@htmlfilenumber` (*Ctr*) Records the number of each HTML file as it is being created. Number 0 is the home page. This might not be sequential, as the user may use `\SetHTMLFileNumber` to create groups of numbered nodes.

```
\newcounter{LWR@htmlfilenumber}
\setcounter{LWR@htmlfilenumber}{0}
```

`\LWR@htmlsectionfilename` {⟨*htmlfilenumber or name*⟩}

Prints the filename for a given section: `\HTMLFilename{}filenumber/name.html`

```
\newcommand*{\LWR@htmlsectionfilename}[1]{%
\LWR@traceinfo{LWR@htmlsectionfilename A !\detokenize{#1}!}%
\begingroup%
```

Disable CJK xpinyin while generating file names.

```
\LWR@disablepinyin%
```

Section 0 or empty is given the home filename. The filename must be detokenized for underscores.

```
% \LWR@traceinfo{about to assign temp}%
\LWR@sanitize{#1}%
\LWR@traceinfo{about to compare with ??}%
\ifdefstring{\LWR@sanitized}{??}
    {\LWR@traceinfo{found ??}}%
    {\LWR@traceinfo{not found ??}}%
\LWR@traceinfo{about to compare with zero or empty}%
\ifboolexpr{
    test {\ifdefstring{\LWR@sanitized}{0}} or
    test {\ifdefstring{\LWR@sanitized}{}} or
    test {\ifdefstring{\LWR@sanitized}{??}}
}
{%
```

```
    \LWR@traceinfo{LWR@htmlsectionfilename B \HomeHTMLFilename.html}%
    \HomeHTMLFilename.html%
}%
```

For a LaTeX section named “Index” or “index” without a prefix, create a filename with a trailing `-0` to avoid colliding with the HTML filename `index.html`:

```
{%
    \LWR@traceinfo{LWR@htmlsectionfilename C \LWR@sanitized}%
    \ifboolexpr{
            test{\ifdefvoid{\HTMLFilename}} and
            (
                test{\ifdefstring{\LWR@sanitized}{Index}} or
                test{\ifdefstring{\LWR@sanitized}{index}}
            )
    }%
    {%
        \LWR@traceinfo{Adding a zero to the index filename.}%
        \LWR@sanitized-0.html%
    }%
```

Otherwise, create a filename with the chosen prefix:

```
    {%
        \HTMLFilename\LWR@isolate{\LWR@sanitized}.html%
    }%
}%
\LWR@traceinfo{LWR@htmlsectionfilename Z}%
\endgroup%
}
```

`\LWR@htmlrefsectionfilename` {⟨*label*⟩}

Prints the filename for the given label

```
\newcommand*{\LWR@htmlrefsectionfilename}[1]{%
    \LWR@traceinfo{LWR@htmlrefsectionfilename: !\detokenize{#1}!}%
```

`\LWR@nullfonts` to allow math in a section name.

```
    \begingroup%
    \LWR@nullfonts%
    \LWR@htmlsectionfilename{\LWR@htmlfileref{#1}}%
    \endgroup%
    \LWR@traceinfo{LWR@htmlrefsectionfilename: done}%
}
```

```
\end{warpHTML}
```

47 Homepage link

for HTML & PRINT:

```
\begin{warpall}
```

`\linkhomename` Holds the default name for the home link.

```
\newcommand{\linkhomename}{Home}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`\LinkHome` May be used wherever you wish to place a link back to the homepage. The filename must be detokenized for underscores.

```
\newcommand*{\LinkHome}{%
    \LWR@subhyperrefclass{\HomeHTMLFilename.html}{\linkhomename}{linkhome}%
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

`\LinkHome` May be used wherever you wish to place a link back to the homepage. For print output, if hyperref is available a hyperlink to the first page is used, named by `\linkhomename`. If hyperref is not available, a pageref is used instead.

`\BaseJobname` is included in the link label in case multiple documents are cross-referenced.

```
\AtBeginDocument{
\@ifundefined{hyperref}{
    \newcommand*{\LinkHome}{%
        \linkhomename\ --- page \pageref{\BaseJobname-page-LWRfirstpage}%
    }
}{
    \newcommand*{\LinkHome}{%
        \hyperref[\BaseJobname-page-LWRfirstpage]{\linkhomename}%
    }
}
}

\AfterEndPreamble{\label{\BaseJobname-page-LWRfirstpage}}
```

```
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@topnavigation` Creates a link to the homepage at the top of the page for use when the window is too narrow for the sideTOC.

```
\newcommand*{\LWR@topnavigation}{%
    \LWR@htmlelementclassline{nav}{topnavigation}{\LinkHome}
}
```

`\LWR@botnavigation` Creates a link to the homepage at the bottom of the page for use when the window is too narrow for the sideTOC.

```
\newcommand*{\LWR@botnavigation}{%
    \LWR@htmlelementclassline{nav}{botnavigation}{\LinkHome}
}
```

```
\end{warpHTML}
```

48 Previous/next navigation links

for HTML & PRINT:
```
\begin{warpall}
```

`\linkpreviousname` What to call the link to the previous web page.

```
\newcommand*{\linkpreviousname}{Previous}
```

`\linknextname` What to call the link to the next web page.

```
\newcommand*{\linknextname}{Next}
```

```
\end{warpall}
```

for PRINT output:
```
\begin{warpprint}
```

`\LinkPrevious` Creates a link to the previous web page if there is one.

```
\newcommand*{\LinkPrevious}{}
```

`\LinkNext` Creates a link to the next web page if there is one.

```
\newcommand*{\LinkNext}{}
```

```
\end{warpprint}
```

for HTML output:
```
\begin{warpHTML}
```

`\LinkPrevious` Creates a link to the previous web page if there is one.

The links refer to the LaTeX labels `\Basejobname-autofile-*`

```
\newcommand*{\LinkPrevious}{%
    \ifnumless{\value{LWR@htmlseqfilenumber}}{1}{}{%
        \setcounter{LWR@tempcountone}{\value{LWR@htmlseqfilenumber}-1}%
        \LWR@subhyperrefclass{%
            \LWR@htmlrefsectionfilename{%
                \BaseJobname-autofile-\arabic{LWR@tempcountone}%
            }%
        }{\linkpreviousname}{linkhome}%
    }%
}
```

`\LinkNext` Creates a link to the next web page if there is one.

The links refer to the LaTeX labels `\Basejobname-autofile-*`
and the last is the label `\Basejobname-autofile-last`

```
\newcommand*{\LinkNext}{%
    \ifcsdef{r@\BaseJobname-autofile-last@lwarp}{%
        \edef\LWR@tempone{%
        \LWR@htmlfileref{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}%
        }%
```

```
        \edef\LWR@temptwo{%
            \LWR@htmlfileref{\BaseJobname-autofile-last}%
        }%
        \ifdefequal{\LWR@tempone}{\LWR@temptwo}{}{%
            \setcounter{LWR@tempcountone}{\value{LWR@htmlseqfilenumber}+1}%
            \LWR@subhyperrefclass{%
                \LWR@htmlrefsectionfilename{%
                    \BaseJobname-autofile-\arabic{LWR@tempcountone}%
                }%
            }{\linknextname}{linkhome}%
        }%
    }{}%
}
```

```
\end{warpHTML}
```

49 \LWRPrintStack diagnostic tool

⚠ Diagnostics tool: Prints the LaTeX nesting depth values for the stack levels. `\LWR@startpars` is used before printing the stack, so that `\LWRPrintStack` may be called from anywhere in the normal text flow.

for HTML output:

```
\begin{warpHTML}
```

`\LWRPrintStack` Prints the closedepth stack.

```
\newcommand*{\LWR@subprintstack}{
\LWR@closedepthone\ \LWR@closedepthtwo\ \LWR@closedepththree\
\LWR@closedepthfour\ \LWR@closedepthfive\ \LWR@closedepthsix\
\LWR@closedepthseven\ \LWR@closedeptheight\ \LWR@closedepthnine\
\LWR@closedepthten\ \LWR@closedeptheleven\ \LWR@closedepthtwelve\
\LWR@closedepththirteen\ \LWR@closedepthfourteen\ \LWR@closedepthfifteen\
\LWR@closedepthsixteen\ \LWR@closedepthseventeen\ \LWR@closedeptheighteen\
\LWR@closedepthnineteen\
}

\newcommand*{\LWRPrintStack}{
\LWR@startpars
\LWR@subprintstack
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}

\newcommand*{\LWRPrintStack}{}

\end{warpprint}
```

50 Closing stack levels

for HTML output:

```
\begin{warpHTML}
```

Close one nested level:

```
\newcommand*{\LWR@closeoneprevious}{%

\LWR@closeone

\LWR@popclose
}
```

`\LWR@closeprevious {⟨sectintype⟩}` Close everything up to the given depth:

```
\newcommand*{\LWR@closeprevious}[1]{%
\LWR@traceinfo{%
    LWR@closeprevious to depth \csuse{LWR@depth#1}, %
    depths are \LWR@subprintstack%
}%
```

Close any pending paragraph:

```
\LWR@stoppars%
```

Close anything nested deeper than the desired depth. First close anything deeper, then at most one of the same level.

```
\whileboolexpr{test{\ifnumcomp{\LWR@closedepthone}{>}{\csuse{LWR@depth#1}}}}%
{%
    \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
    \LWR@closeoneprevious%
}%
\ifboolexpr{test{\ifnumcomp{\LWR@closedepthone}{=}{\csuse{LWR@depth#1}}}}%
{%
    \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
    \LWR@closeoneprevious%
}{}%
\LWR@traceinfo{LWR@closeprevious: done, depths are \LWR@subprintstack}%
}

\end{warpHTML}
```

51 PDF pages and styles

for HTML output:
```
\begin{warpHTML}
```

`\LWR@forcenewpage` New PDF page a before major environment.

This is used just before major environments, such as `verse`. Reduces the chance of an environment overflowing the HTML PDF output page.

```
\newcommand{\LWR@forcenewpage}{%
\LWR@traceinfo{LWR@forcenewpage}%
\ifinner\else%
    \LWR@traceinfo{LWR@forcenewpage A}%
    \LWR@stoppars%
    \LWR@traceinfo{LWR@forcenewpage B}%
    \LWR@maybe@orignewpage%
```

```
    \LWR@traceinfo{LWR@forcenewpage C}%
    \LWR@startpars%
\fi%
\LWR@traceinfo{LWR@forcenewpage done}%
}
```

\pagestyle, etc. are nullified for HTML output.

\pagestyle {⟨*style*⟩}

```
\renewcommand*{\pagestyle}[1]{}
```

\thispagestyle {⟨*style*⟩}

```
\renewcommand*{\thispagestyle}[1]{}
```

\markboth {⟨*left*⟩} {⟨*right*⟩}

```
\renewcommand*{\markboth}[2]{}
```

\markright {⟨*right*⟩}

```
\renewcommand*{\markright}[1]{}
```

\raggedbottom

```
\renewcommand*{\raggedbottom}{}
```

\flushbottom

```
\renewcommand*{\flushbottom}{}
```

\sloppy

```
\renewcommand*{\sloppy}{}
```

\fussy

```
\renewcommand*{\fussy}{}
```

\pagenumbering * {⟨*commands*⟩}

```
\RenewDocumentCommand{\pagenumbering}{s m}{}
\end{warpHTML}
```

52 HTML tags, spans, divs, elements

for HTML output:

```
\begin{warpHTML}
```

52.1 Mapping LaTeX sections to HTML sections

```
\newcommand*{\LWR@tagtitle}{h1}
\newcommand*{\LWR@tagtitleend}{/h1}
\newcommand*{\LWR@tagbook}{div class=\textquotedbl{}book\textquotedbl}
\newcommand*{\LWR@tagbookend}{/div}
\newcommand*{\LWR@tagpart}{h2}
\newcommand*{\LWR@tagpartend}{/h2}
\newcommand*{\LWR@tagchapter}{h3}
\newcommand*{\LWR@tagchapterend}{/h3}
\newcommand*{\LWR@tagsection}{h4}
\newcommand*{\LWR@tagsectionend}{/h4}
\newcommand*{\LWR@tagsubsection}{h5}
\newcommand*{\LWR@tagsubsectionend}{/h5}
\newcommand*{\LWR@tagsubsubsection}{h6}
\newcommand*{\LWR@tagsubsubsectionend}{/h6}
\newcommand*{\LWR@tagparagraph}{span class=\textquotedbl{}paragraph\textquotedbl}
\newcommand*{\LWR@tagparagraphend}{/span}
\newcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
\newcommand*{\LWR@tagsubparagraphend}{/span}

\newcommand*{\LWR@tagregularparagraph}{p}
```

52.2 Hook while processing tags

`\LWR@hook@processingtags` (*Hook*) [lwarp]

This is used to disable special text processing while processing HTML tags. Special processing includes that done by babel-french, luavina, xevlna.

`\LWR@hook@processingtags`

Disable special text processing while generating tags. Replaces `\LWR@FBcancel` in most places.

```
\newcommand*{\LWR@hook@processingtags}{}
```

52.3 Babel-French tag modifications

Adjust babel-french for HTML spaces. So far, this only works for *pdflatex* and *xelatex*.

(*Emulates or patches code by* Daniel Flipo.)

```
\providecommand*{\LWR@FBcancel}{}

\AtBeginDocument{%
```

In some circumstances, `\NoAutoSpacing` may be defined when `\frenchbsetup` is not.

```
\@ifundefined{NoAutoSpacing}%
    {}%
    {%
        \LetLtxMacro\LWR@FBcancel\NoAutoSpacing%
        \appto{\LWR@hook@processingtags}{\LWR@FBcancel}%
    }%

\@ifundefined{frenchbsetup}%
```

```
{}%
{%
    \frenchbsetup{FrenchFootnotes=false}%

%
    \renewrobustcmd*{\FBcolonspace}{%
        \begingroup%
        \LWR@hook@processingtags%
        \LWR@origampersand{}nbsp;%
        \endgroup%
    }%
    \renewrobustcmd*{\FBthinspace}{%
        \begingroup%
        \LWR@hook@processingtags%
        \LWR@origampersand\LWR@origpound{}x202f;% \,
        \endgroup%
    }%
    \renewrobustcmd*{\FBguillspace}{%
        \begingroup%
        \LWR@hook@processingtags%
        \LWR@origampersand{}nbsp;% ~,  for \og xyz \fg{}
        \endgroup%
    }%
    \DeclareDocumentCommand{\FBmedkern}{}{%
        \begingroup%
        \LWR@hook@processingtags%
        \LWR@origampersand\LWR@origpound{}x202f;% \,
        \endgroup%
    }%
    \DeclareDocumentCommand{\FBthickkern}{}{%
        \begingroup%
        \LWR@hook@processingtags%
        \LWR@origampersand{}nbsp;% ~
        \endgroup%
    }%
    \renewrobustcmd*{~}{\HTMLentity{nbsp}}% was overwritten by babel-french
    \ifFBunicode%
    \else%
        \DeclareTextSymbol{\FBtextellipsis}{LY1}{133}%
        \DeclareTextCommandDefault{\FBtextellipsis}{\textellipsis\xspace}%
    \fi%
}%
}
```

52.4 HTML output formatting

Helps format the output HTML code for human readability.

`\LWR@indentHTML` Newline and indent the output HTML code.

```
\newcommand*{\LWR@indentHTML}{%
    \LWR@orignewline\LWR@origrule{2em}{0pt}%
}
```

`\LWR@indentHTMLtwo` Newline and indent the output HTML code.

```
\newcommand*{\LWR@indentHTMLtwo}{%
    \LWR@orignewline\LWR@origrule{4em}{0pt}%
}
```

52.5 HTML tags

`\LWR@htmltagc {⟨tag⟩}` Break ligatures and use upright apostrophes in HTML tags.

`\protect` is in case the tag appears in TOC, LOF, LOT.

```
\newcommand*{\LWR@htmltagc}[1]{%
    \LWR@traceinfo{LWR@htmltagc !\detokenize{#1}!}%
    \begingroup%
    \LWR@hook@processingtags%
    \LWR@fontfortags{LWR@htmltagc}{\detokenize{#1}}%
    \protect\LWR@origtextless%
    \LWR@isolate{#1}%
    \protect\LWR@origtextgreater%
    \endgroup%
}
```

`\LWR@spanwarnformat {⟨object⟩}`

Warns if the given object is used inside a span.

```
\newcommand*{\LWR@spanwarnformat}[1]{%
    \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
        \PackageWarning{lwarp}{%
            A #1 is being used inside a span.\MessageBreak
            Formatting may be lost,%
        }%
    }{}%
}
```

`\LWR@spanwarninvalid {⟨object⟩}`

Warns if the given object is used inside a span.

```
\newcommand*{\LWR@spanwarninvalid}[1]{%
    \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
        \PackageWarning{lwarp}{%
            A #1 is being used inside a span.\MessageBreak
            This generates invalid HTML,%
        }%
    }{}%
}
```

`LWR@nestspan` (*env.*) Disable `minipage`, `\parbox`, and HTML `<div>`s inside a `<span>`.

⚠ `\begin{LWR@nestspan}` must follow the opening `<span>` tag to allow a paragraph to start if the span is at the beginning of a new paragraph.

⚠ `\end{LWR@nestspan}` must follow the `</span>` or a `<p>` may appear inside the span.

```
\newcommand*{\LWR@nestspanitem}{%
    \if@newlist\else{
        \LWR@htmltagc{br /}%
        \LWR@orignewline%
    }\fi%
    \LWR@origitem%
}

\newenvironment*{LWR@nestspan}
{%
    \LWR@traceinfo{LWR@nestspan starting}%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
    {%
        \LWR@traceinfo{LWR@nestspan: inside a lateximage}%
    }%
    {% not in a lateximage
        \LWR@traceinfo{LWR@nestspan: NOT inside a lateximage}%
        \addtocounter{LWR@spandepth}{1}%
```

Nullify several objects inside the span:

```
        \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}%
            {\LWR@spanwarnformat{minipage or \protect\parbox}}%
            {}%
        \RenewDocumentEnvironment{BlockClass}{o m}%
            {\LWR@spanwarnformat{multi-paragraph object}}%
            {}%
        \RenewDocumentEnvironment{LWR@BlockClassWP}{m m D(){} m}%
            {\LWR@spanwarnformat{multi-paragraph object}}%
            {}%
        \renewcommand{\BlockClassSingle}[2]{%
            {\LWR@spanwarnformat{multi-paragraph object}}%
            ##2%
        }%
        \renewcommand{\LWR@forcenewpage}{}%
        \renewcommand{\LWR@liststart}{\LetLtxMacro\item\LWR@nestspanitem}%
        \renewcommand{\LWR@listend}{\leavevmode}%
        \renewenvironment{quote}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
     \renewenvironment{quotation}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
    }% not in a lateximage
    \LWR@traceinfo{LWR@nestspan starting: done}%
}% starting env
{% ending env
    \LWR@traceinfo{LWR@nestspan ending}%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
    {}%
    {\addtocounter{LWR@spandepth}{-1}}%
    \LWR@traceinfo{LWR@nestspan ending: done}%
}
```

`\LWR@htmlspan` {⟨*tag*⟩} {⟨*text*⟩}

`\LWR@spandepth` is used to ensure that paragraph tags are not generated inside a span. The exact sequence of when to add and subtract the counter is important to correctly handle the paragraph tags before and after the span.

```
\NewDocumentCommand{\LWR@htmlspan}{m +m}{%
    \LWR@ensuredoingapar%
    \LWR@htmltagc{#1}%
```

```
6409     \begin{LWR@nestspan}%
6410     #2%
6411     \LWR@htmltagc{/#1}%
6412     \end{LWR@nestspan}%
6413 }
```

\LWR@htmlspanclass [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩} {⟨*text*⟩}

```
6414 \NewDocumentCommand{\LWR@htmlspanclass}{o D(){} m +m}{%
6415     \LWR@traceinfo{LWR@htmlspanclass |#1|#2|#3|}%
6416     \LWR@ensuredoingapar%
6417     \ifblank{#2}%
6418         {\LWR@subhtmlelementclass{span}[#1]{#3}}%
6419         {\LWR@subhtmlelementclass{span}[#1](#2){#3}}%
6420     \begin{LWR@nestspan}%
6421     #4%
6422     \LWR@htmltagc{/span}%
6423     \LWR@traceinfo{LWR@htmlspanclass done}%
6424     \end{LWR@nestspan}%
6425 }
```

\LWR@htmltag {⟨*tag*⟩}

Print an HTML tag: <tag>

```
6426 \newcommand*{\LWR@htmltag}[1]{%
6427     \LWR@htmltagc{#1}%
6428 }
```

52.6 Block tags and comments

In the following, \origttfamily breaks ligatures, which may not be used for HTML codes:

\LWR@htmlopencomment
\LWR@htmlclosecomment

```
6429 \newcommand*{\LWR@htmlopencomment}{%
6430 % \LWR@traceinfo{LWR@htmlopencomment}%
6431     \begingroup%
6432     \LWR@hook@processingtags%
6433     \LWR@fontfortags{LWR@htmlopencomment}{}%
6434     \LWR@print@mbox{\LWR@origtextless{}!-\/-}%
6435     \endgroup%
6436 }
6437
6438 \newcommand*{\LWR@htmlclosecomment}{%
6439 % \LWR@traceinfo{LWR@htmlclosecomment}%
6440     \begingroup%
6441     \LWR@hook@processingtags%
6442     \LWR@fontfortags{LWr@htmlclosecomment}{}%
6443     \LWR@print@mbox{-\/-\LWR@origtextgreater}%
6444     \endgroup%
6445 }
```

\LWR@htmlcomment {⟨*comment*⟩}

```
\newcommand{\LWR@htmlcomment}[1]{%
    \ifmmode%
    \else%
        \LWR@htmlopencomment{}%
        {%
            \LWR@print@normalfont%
            \LWR@origttfamily% break ligatures
            #1%
        }%
        \LWR@htmlclosecomment{}%
    \fi%
}
```

`\LWR@htmlblockcomment {⟨comment⟩}`

```
\newcommand{\LWR@htmlblockcomment}[1]
    {\LWR@stoppars\LWR@htmlcomment{#1}\LWR@startpars}
```

`\LWR@htmlblocktag {⟨tag⟩}` print a stand-alone HTML tag

```
\newcommand*{\LWR@htmlblocktag}[1]{%
    \LWR@stoppars%
    \LWR@htmltag{#1}%
    \LWR@startpars%
}
```

52.7 Div class and element class

`\LWR@subhtmlelementclass {⟨element⟩} [⟨style⟩] (⟨aria role⟩) {⟨class⟩}`

Factored and reused in several places.

The trailing spaces allow more places for a line break.

The use of `\textquotedbl` instead of " provides improved compatibility with xeCJK.

```
\NewDocumentCommand{\LWR@subhtmlelementclass}{m O{} D(){} m}{%
    \LWR@traceinfo{LWR@subhtmlelementclass !#1!#2!#3!#4!}%
    \ifblank{#2}%
    {% empty style
        \LWR@htmltag{%
            #1%
            \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}% spaces
            \ifblank{#4}{}{ class=\textquotedbl#4\textquotedbl}% spaces
        }%
    }%
    {% non-empty style
        \LWR@htmltag{%
            #1\LWR@indentHTML%
          \ifblank{#3}{}{role=\textquotedbl#3\textquotedbl\LWR@indentHTML}%
         \ifblank{#4}{}{class=\textquotedbl#4\textquotedbl\LWR@indentHTML}%
            style=\textquotedbl#2\textquotedbl\LWR@orignewline%
        }%
    }%
    \LWR@traceinfo{LWR@subhtmlelementclass done}%
}
```

`\LWR@htmlelementclass` {⟨*element*⟩} [⟨*style*⟩] {⟨*class*⟩}

```
\NewDocumentCommand{\LWR@htmlelementclass}{m o D(){} m}{%
    \LWR@stoppars%
    \LWR@forceemptyline%
    \ifblank{#3}%
        {\LWR@subhtmlelementclass{#1}[#2]{#4}}%
        {\LWR@subhtmlelementclass{#1}[#2](#3){#4}}%
    \LWR@startpars%
}
```

`\LWR@htmlelementclassend` {⟨*element*⟩} {⟨*class*⟩}

```
\newcommand*{\LWR@htmlelementclassend}[2]{%
    \LWR@stoppars%
    \LWR@htmltag{/#1}%
    \ifbool{HTMLDebugComments}{%
        \LWR@htmlcomment{End of #1 ``#2''}%
    }{}%
    \LWR@startpars%
}
```

`\LWR@htmldivclass` [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩}

```
\NewDocumentCommand{\LWR@htmldivclass}{o D(){} m}{%
    \ifblank{#2}
        {\LWR@htmlelementclass{div}[#1]{#3}}%
        {\LWR@htmlelementclass{div}[#1](#2){#3}}%
}
```

`\LWR@htmldivclassend` {⟨*class*⟩}

```
\newcommand*{\LWR@htmldivclassend}[1]{%
    \LWR@htmlelementclassend{div}{#1}%
}
```

52.8 Single-line elements

A single-line element, without a paragraph tag for the line of text:

`\LWR@htmlelementclassline` {⟨*element*⟩} [⟨*style*⟩] {⟨*class*⟩} {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@htmlelementclassline}{m o m +m}{%
    \LWR@stoppars
    \LWR@forceemptyline%
    \LWR@subhtmlelementclass{#1}[#2]{#3}%
    #4%
    \LWR@htmltag{/#1}
    \LWR@startpars
}
```

52.9 HTML5 semantic elements

`\LWR@htmlelement` {⟨*element*⟩}

```
\newcommand*{\LWR@htmlelement}[1]{%
    \LWR@htmlblocktag{#1}
}
```

`\LWR@htmlelementend` {⟨*element*⟩}

```
\newcommand*{\LWR@htmlelementend}[1]{%
    \LWR@stoppars
    \LWR@htmltag{/#1}
    \LWR@startpars
}

\end{warpHTML}
```

52.10 High-level block and inline classes

These are high-level commands which allow the creation of arbitrary block or inline sections which may be formatted with css.

Nullified versions are provided for print mode.

For other direct-formatting commands, see section 95.

`BlockClass` (*env.*) `[`⟨*style*⟩`]` `(`⟨*aria role*⟩`)` {⟨*class*⟩} High-level interface for `<div>` classes.

Ex: `\begin{BlockClass}{class}` text `\end{BlockClass}`

for HTML & PRINT:

```
\begin{warpall}
\NewDocumentEnvironment{BlockClass}{o D(){} m}{}{}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}

\NewDocumentEnvironment{LWR@HTML@BlockClass}{o D(){} m}%
    {\LWR@htmldivclass[#1](#2){#3}}%
    {\LWR@htmldivclassend{#3}}

\LWR@formattedenv{BlockClass}
\end{warpHTML}
```

`\BlockClassSingle` {⟨*class*⟩} {⟨*text*⟩} A single-line `<div>`, without a paragraph tag for the line of text.

for HTML & PRINT:

```
\begin{warpall}
\newcommand{\BlockClassSingle}[2]{#2}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
\newcommand{\LWR@HTML@BlockClassSingle}[2]{%
    \LWR@htmlelementclassline{div}{#1}{#2}%
}
```

```
6545
6546 \LWR@formatted{BlockClassSingle}
6547 \end{warpHTML}
```

`\InlineClass` (⟨*WP style*⟩) [⟨*style*⟩] {⟨*class*⟩} {⟨*text*⟩}

High-level interface for inline span classes.

(⟨*WP style*⟩) is css styling to add when formatting for a word processor import.

[⟨*style*⟩] is the css styling to add when not formatting for a word processor.

for HTML & PRINT:

```
6548 \begin{warpall}
6549 \NewDocumentCommand{\InlineClass}{D{(}{)}{} o m +m}{#4}%
6550 \end{warpall}
```

for HTML output:

```
6551 \begin{warpHTML}
6552 \NewDocumentCommand{\LWR@HTML@InlineClass}{D{(}{)}{} o m +m}{%
6553     \LWR@traceinfo{LWR@HTML@InlineClass #3}%
6554     \ifbool{FormatWP}{%
6555         \LWR@traceinfo{LWR@HTML@InlineClass: FormatWP}%
6556         \LWR@htmlspanclass[#1]{#3}{#4}%
6557     }{%
6558         \LWR@traceinfo{LWR@HTML@InlineClass: not FormatWP}%
6559         \LWR@htmlspanclass[#2]{#3}{#4}%
6560     }%
6561     \LWR@traceinfo{LWR@HTML@InlineClass: done}%
6562 }
6563
6564 \LWR@formatted{InlineClass}
6565 \end{warpHTML}
```

`LWR@BlockClassWP` (*env.*) {⟨*WPstyle*⟩} {⟨*HTMLstyle*⟩} (⟨*aria role*⟩) {⟨*class*⟩} Low-level interface for `<div>` classes with an automatic float ID. These are often used when `\ifbool{FormatWP}`.

The use of `\textquotedbl` instead of " provides improved compatibility with xeCJK.

for HTML & PRINT:

```
6566 \begin{warpall}
6567 \NewDocumentEnvironment{LWR@BlockClassWP}{m m D(){} m}{}{}
6568 \end{warpall}
```

for HTML output:

```
6569 \begin{warpHTML}
6570 \NewDocumentEnvironment{LWR@HTML@LWR@BlockClassWP}{m m D(){} m}%
6571     {%
6572         \LWR@stoppars%
6573         \ifbool{FormatWP}%
6574         {%
6575             \addtocounter{LWR@thisautoidWP}{1}%

6576             \LWR@htmltag{%
6577                 div class=\textquotedbl#4\textquotedbl\ % space
6578                 id=\textquotedbl%
6579                     \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
6580                 \textquotedbl%
6581                 \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}%
6582                 \ifblank{#1}{}{ style=\textquotedbl#1\textquotedbl}%
6583             }%
```

```
        }% FormatWP
        {% not FormatWP
            \LWR@htmltag{%
                div class=\textquotedbl#4\textquotedbl%
                \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}%
                \ifblank{#2}{}{ style=\textquotedbl#2\textquotedbl}%
            }%
        }% not FormatWP
        \LWR@startpars%
    }
    {\LWR@htmldivclassend{#4}}

\LWR@formattedenv{LWR@BlockClassWP}
\end{warpHTML}
```

52.11 Closing HTML tags

for HTML output:

```
\begin{warpHTML}
```

Sections H1, H2, etc. do not need a closing HTML tag, but we add a comment for readability:

```
\newcommand*{\LWR@printclosebook}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing book}}{}}
\newcommand*{\LWR@printclosepart}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing part}}{}}
\newcommand*{\LWR@printclosechapter}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing chapter}}{}}
\newcommand*{\LWR@printclosesection}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing section}}{}}
\newcommand*{\LWR@printclosesubsection}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsection}}{}}
\newcommand*{\LWR@printclosesubsubsection}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsubsection}}{}}
\newcommand*{\LWR@printcloseparagraph}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing paragraph}}{}}
\newcommand*{\LWR@printclosesubparagraph}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subparagraph}}{}}
```

Lists require closing HTML tags:

```
\newcommand*{\LWR@printcloselistitem}
    {\LWR@htmltag{/li}}
\newcommand*{\LWR@printclosedescitem}
    {\LWR@htmltag{/dd}}
\newcommand*{\LWR@printcloseitemize}
    {\LWR@htmltag{/ul}}
\newcommand*{\LWR@printcloseenumerate}
    {\LWR@htmltag{/ol}}
\newcommand*{\LWR@printclosedescription}
    {\LWR@htmltag{/dl}}


\end{warpHTML}
```

53 Paragraph handling

These commands generate the HTML paragraph tags when allowed and required.

Paragraph tags are or are not allowed depending on many conditions. Section 54 has high-level commands which allow paragraph-tag generation to start/stop. Even when allowed (LWR@doingstartpars), tags are not generated until a LaTeX paragraph is being used (LWR@doingapar). LWR@lateximagedepth is used to prevent nesting tags inside a lateximage. LWR@spandepth is used to prevent nesting paragraph tags inside a paragraph, which became important inside \fbox commands and other spans.

The LaTeX paragraph hooks are used to manage tag creation.

for HTML output:
```
\begin{warpHTML}
```

LWR@spandepth (*Ctr*) Do not create paragraph tags inside of an HTML span.

```
\newcounter{LWR@spandepth}
\setcounter{LWR@spandepth}{0}
```

LWR@doingparhooks (*bool*) Tells whether the lwarp paragraph hooks are to be active.

```
\newbool{LWR@doingparhooks}
\boolfalse{LWR@doingparhooks}
```

LWR@in@multirow@par (*bool*) Tells whether to generate break instead of paragraph tags inside a \multirow.

```
\newbool{LWR@in@multirow@par}
\boolfalse{LWR@in@multirow@par}
```

LWR@starting@fancybox (*bool*) Suppresses <br> if beginning a fancybox environment.

```
\newbool{LWR@starting@fancybox}
\boolfalse{LWR@starting@fancybox}
```

LWR@doingstartpars (*bool*) Tells whether paragraphs may be generated.

```
\newbool{LWR@doingstartpars}
\boolfalse{LWR@doingstartpars}
```

LWR@doingapar (*bool*) Tells whether have actually generated and are currently processing paragraph text.

```
\newbool{LWR@doingapar}
\global\boolfalse{LWR@doingapar}
```

LWR@algocf@dopars (*bool*) Tells whether algorithm2e has patched paragraph handling using \everypar. If so, the open paragraph tags are generated by algorithm2e's \algocf@everypar instead of \LWR@openparagraph.

```
\newbool{LWR@algocf@dopars}
\boolfalse{LWR@algocf@dopars}
```

\PN@parnotes@auto Redefined by parnotes to print paragraph notes at the end of each paragraph.

```
\def\PN@parnotes@auto{}%
```

\LWR@ensuredoingapar
\LWR@openparagraph

These were different in older versions of lwarp, but are now the same thing.

```
\newcommand*{\LWR@openparagraph}
{%
```

See if paragraph handling is enabled:

```
    \ifboolexpr{
        bool{LWR@doingparhooks} and
        bool{LWR@doingstartpars}
    }%
    {% handling pars
```

See if have already started a `lateximage` or a . If so, do not generate nested paragraph tags.

```
        \ifboolexpr{
            test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
            test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
        }% nested par tags?
```

If so: Do nothing if already started a `lateximage` page. Cannot nest a `lateximage`. Also do nothing if already inside a . Do not nest paragraph tags inside a .

```
        {}% no nested par tags
```

Else: No `lateximage` or has been started yet, so it's OK to generate paragraph tags.

```
        {% yes nest par tags
            \ifbool{LWR@doingapar}{}{%
```

If parnotes is used, paragraph notes are inserted before starting the next paragraph:

```
                \PN@parnotes@auto%
```

Set flag before creating the tag, so that the tag itself does not trigger a new paragraph:

```
                \global\booltrue{LWR@doingapar}%
```

The opening paragraph tag. Do not create tag if doing algorithm2e handling instead:

```
                \ifbool{LWR@algocf@dopars}{}{%
                    \ifbool{LWR@in@multirow@par}%
                        {}%
                  {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
                }%
            }%
        }% end of yes nest par tags
    }% end of handling pars
    {}% not handling pars
}

\let\LWR@ensuredoingapar\LWR@openparagraph
```

\LWR@closeparagraph@br Add an HTML break if in a span, and not in a lateximage, and not in tabular metadata. Factored from \LWR@closeparagraph.

```
\newcommand*{\LWR@closeparagraph@br}
{%
    \ifboolexpr{
        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} and
        test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}} and
        not bool {LWR@starting@fancybox} and
        not bool {LWR@intabularmetadata} or
        bool {LWR@in@multirow@par}
    }%
        {\unskip\LWR@htmltagc{br /}}%
        {}%
}
```

\LWR@closeparagraph

```
\newcommand*{\LWR@closeparagraph}
{%
% \LWR@traceinfo{LWR@closeparagraph}%
```

See if paragraph handling is enabled:

```
    \ifbool{LWR@doingparhooks}{%
        \ifbool{LWR@doingapar}%
```

If currently in paragraph mode:

```
        {% handling pars
```

See if already started a lateximage or a :

```
            \ifboolexpr{
                test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
                test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} or
                bool{LWR@in@multirow@par}
            }%
```

Add a parbreak if in a span, not in a lateximage, and not in table metadata.

```
            {% no nested par tags
                \LWR@closeparagraph@br%
            }% no nested par tags
```

If have not already started a lateximage or a :

```
            {% yes nest par tags
```

Print a closing tag.

(The fill seems to be required to force the caption package to create flush left caption text in the HTML.)

```
                \@hspacer{\fill}% \hspace*{\fill}
                \leavevmode\LWR@orignewline%
                \LWR@htmltagc{/\LWR@tagregularparagraph}%
```

No longer doing a paragraph:

```
                \global\boolfalse{LWR@doingapar}%
```

Disable the special `minipage` & `\hspace` interaction until a new minipage is found:

```
                \global\boolfalse{LWR@minipagethispar}%
```

If parnotes is used, paragraph notes are inserted after ending the previous paragraph:

```
                \PN@parnotes@auto%

            }% end of yes nest par tags
        }% LWR@doingapar: end of handling pars
```

Add a parbreak if in a span, not in a lateximage, and not in table metadata.

```
        {% not LWR@doingapar: not handling pars
            \LWR@closeparagraph@br%
        }% not handling pars
```

In most cases, finish with a LaTeX `\par`, but in the case of paragraphs between lines in a tabular fetch the next token instead. Required for `\multicolumn`.

```
        \ifboolexpr{%
            not bool {LWR@doingapar} and
            test {\ifnumcomp{\value{LWR@tabulardepth}}{>}{0}} and
            test {
           \ifnumcomp{\value{LWR@tabulardepth}}{=}{\value{LWR@tabularpardepth}}
            } and
            bool {LWR@intabularmetadata} and
            not bool {LWR@tableparcell} and
            test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}}
        }%
            {\LWR@getmynexttoken}%
            {}%
    }% LWR@doingparhooks
    {}% not LWR@doingparhooks
% Do not place anything here, due to the above \LWR@getmynexttoken.
}
```

53.1 Paragraph Hooks

para/begin (*Hook*) [LaTeX]

```
\AddToHook{para/begin}[lwarp]{\LWR@openparagraph}
```

para/end (*Hook*) [LaTeX]

```
\AddToHook{para/end}[lwarp]{\LWR@closeparagraph}
```

```
\end{warpHTML}
```

54 Paragraph start/stop handling

These commands allow/disallow the generation of HTML paragraph tags.

Section 53 has the commands which actually generate the tags.

The LaTeX paragraph hooks are used to generate the opening and closing paragraph tags.

for HTML output:
```
\begin{warpHTML}
```

`\LWR@startpars` Begin handling HTML paragraphs. This allows an HTML paragraph to start, but one has not yet begun.

```
\newcommand*{\LWR@startpars}%
{%
```

Ignore if inside a `lateximage` or :

```
    \ifboolexpr{
        test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
    }%
        {}% nesting
        {% not nesting
```

The LaTeX paragraph hook controls tag generation for the start and end of paragraphs.

See if currently handling HTML paragraphs:

```
         \ifboolexpr {bool{LWR@doingparhooks} and bool{LWR@doingstartpars}}%
```

If already in paragraph mode, do nothing.

```
                {}%
```

If not currently in paragraph mode:

```
                {\par}%
```

Are now handling paragraphs, but have not yet actually started one:

```
            \global\booltrue{LWR@doingstartpars}%
```

No <par> tag yet to undo:

```
            \global\boolfalse{LWR@doingapar}%
        }% not nesting
}
```

`\LWR@stoppars` Stop handling HTML paragraphs. Any currently open HTML paragraph is closed, and no more will be opened.

```
\newcommand*{\LWR@stoppars}%
{%
```

Ignore if inside a lateximage or :

```
    \ifboolexpr{
        test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
    }%
        {}% nesting
        {% not nesting
```

See if currently handling HTML paragraphs:

```
            \ifboolexpr{bool{LWR@doingparhooks} and bool{LWR@doingapar}}%
```

if currently in an HTML paragraph:

```
            {%
```

Print a closing tag:

```
                \leavevmode\LWR@orignewline%
                \LWR@htmltagc{/\LWR@tagregularparagraph}%
                \LWR@orignewline%
```

No longer have an open HTML paragraph:

```
                \global\boolfalse{LWR@doingapar}%
```

Disable the special minipage & \hspace interaction until a new minipage is found:

```
                \global\boolfalse{LWR@minipagethispar}%
            }%
```

If was not in an HTML paragraph:

```
            {}%
```

No longer in paragraph mode:

```
            \global\setbool{LWR@doingstartpars}{false}%
```

No <p> tag to undo:

```
            \global\boolfalse{LWR@doingapar}%
        }% not nesting
}

\end{warpHTML}
```

55 Indentfirst

indentfirst (*Pkg*) indentfirst redefines \@afterindentfalse to be \@afterindenttrue. This is reversed \AtBeginDocument here.

for HTML output:

```
\begin{warpHTML}
```

```
\AtBeginDocument{
    \def\@afterindentfalse{\let\if@afterindent\iffalse}
    \@afterindentfalse
}
\let\LWR@afterindent@syntaxhighlight\fi% syntax highlighting

\end{warpHTML}
```

56 Page headers and footers

for HTML & PRINT:

```
\begin{warpall}
```

In the following, catcode is manually changed back and forth without groups, since new macros are being defined which must not be contained within the groups.

```
\newcommand{\LWR@firstpagetop}{} % for the home page alone
\newcommand{\LWR@firstpagebottom}{} % for the home page alone
\newcommand{\LWR@pagetop}{} % for all other pages
\newcommand{\LWR@pagebottom}{}
```

`\HTMLFirstPageTop` {⟨*text and logos*⟩}

```
\newcommand{\HTMLFirstPageTop}[1]{%
    \renewcommand{\LWR@firstpagetop}{#1}%
}
```

`\HTMLFirstPageBottom` {⟨*text and logos*⟩}

```
\newcommand{\HTMLFirstPageBottom}[1]{%
    \renewcommand{\LWR@firstpagebottom}{#1}%
}
```

`\HTMLPageTop` {⟨*text and logos*⟩}

```
\newcommand{\HTMLPageTop}[1]{%
    \renewcommand{\LWR@pagetop}{#1}%
}
```

`\HTMLPageBottom` {⟨*text and logos*⟩}

```
\newcommand{\HTMLPageBottom}[1]{%
    \renewcommand{\LWR@pagebottom}{#1}%
}

\end{warpall}
```

57 CSS

for HTML output:

```
\begin{warpHTML}
```

`\LWR@currentcss` The css filename to use. This may be changed mid-document using `\CSSFilename`, allowing different css files to be used for different sections of the document.

```
\newcommand*{\LWR@currentcss}{lwarp.css}
```

`\CSSFilename` {⟨*new-css-filename.css*⟩} Assigns the css file to be used by the following HTML pages.

```
\newcommand*{\CSSFilename}[1]{%
    \renewcommand*{\LWR@currentcss}{#1}%
    \@onelevel@sanitize\LWR@currentcss%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\CSSFilename}[1]{}
\end{warpprint}
```

58 MathJax script

for HTML output:

```
\begin{warpHTML}
```

Default: `lwarp_mathjax.txt`

`\LWR@mathjaxfilename` The MathJax script filename to use. This file is copied into the head of each HTML page. This may be changed mid-document using `\MathJaxFilename`, allowing the use of a custom MathJax script, such as for a local repository, or different MathJax script files to be used for different sections of the document.

```
\newcommand*{\LWR@mathjaxfilename}{lwarp_mathjax.txt}
```

`\MathJaxFilename` {⟨*filename*⟩} Assigns the MathJax script file to be used by the following HTML pages.

```
\newcommand*{\MathJaxFilename}[1]{%
    \renewcommand*{\LWR@mathjaxfilename}{#1}%
    \@onelevel@sanitize\LWR@mathjaxfilename%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\MathJaxFilename}[1]{}
\end{warpprint}
```

59 Title, HTML meta author, HTML meta description

for HTML output:

```
\begin{warpHTML}
```

`\title` {⟨*title*⟩} Modified to remember `\thetitle`, which is used to set the HTML page titles.

```
\let\LWR@origtitle\title

\renewcommand*{\title}[1]{%
    \LWR@origtitle{#1}%
    \begingroup%
        \renewcommand{\thanks}[1]{}%
        \protected@xdef\thetitle{#1}%
    \endgroup%
}

\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

`\HTMLTitle` {⟨*Titlename*⟩} The Title to place into an HTML meta tag. The default is to use the document `\title`'s setting.

```
\providecommand{\thetitle}{\BaseJobname}

\newcommand{\theHTMLTitle}{\thetitle}

\newcommand{\HTMLTitle}[1]{\renewcommand{\theHTMLTitle}{#1}}
```

`\HTMLAuthor` {⟨*authorname*⟩} The author to place into an HTML meta tag. If none given, the default is `\theauthor`, which is empty unless the titling package is used.

```
\providecommand{\theauthor}{}

\newcommand{\theHTMLAuthor}{\theauthor}

\newcommand{\HTMLAuthor}[1]{\renewcommand{\theHTMLAuthor}{#1}}
```

This is placed inside an HTML meta tag at the start of each file. This may be changed mid-document using `\HTMLDescription`, allowing different HTML descriptions to be used for different sections of the document.

⚠ HTML author Do not use double quotes, and do not exceed 150 characters.

`\HTMLDescription` {⟨*New html meta description.*⟩} Assigns the HTML file's description meta tag.

```
\newcommand{\LWR@currentHTMLDescription}{}

\newcommand{\HTMLDescription}[1]{%
\renewcommand{\LWR@currentHTMLDescription}{#1}
}

\end{warpall}
```

60 Footnotes

lwarp uses native LaTeX footnote code, although with its own `\box` to avoid the LaTeX output routine. The usual functions mostly work as-is.

footnote numbering To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
— or —
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

MathJax Also for MathJax, \footnotename is used for a \footnotemark if the actual footnote number is not known. To redefine it, provide it before loading lwarp:

```
\providecommand{\footnotename}{something}
\usepackage{lwarp}
```

Similar for sidenotes. For endnotes:

```
\def\endnotename{something}% \def allows name to start with
"end"
```

For the pagenote package, there is no \pagenotename to define, since there is no \pagenotemark command.

footmisc The footmisc stable option is emulated by lwarp.

⚠ sectioning commands When using footnotes in sectioning commands, to generate consistent results between print and HTML, use the footmisc package with the stable option, provide a short TOC entry, and \protect the \footnote:

```
\usepackage[stable]{footmisc}
. . .
\subsection[Subsection Name]
    {Subsection Name\protect\footnote{A footnote.}}
```

memoir with footmisc ⚠ memoir If using memoir class, with which lwarp preloads footmisc, the stable option must be declared before lwarp is loaded:

```
\PassOptionsToPackage{stable}{footmisc}
\usepackage{lwarp}
. . .
```

Do not use a starred sectioning command. As an alternative, it may be possible to adjust \secnumdepth instead.

Several kinds of footnotes are used: in a regular page, in a minipage, or as thanks in the titlepage. Each of these is handle differently.

60.1 Regular page footnotes

In HTML documents, footnotes are placed at the bottom of the web page or the section, depending on FootnoteDepth, using the LaTeX box \LWR@footnotebox. Using this instead of the original \footins box avoids having footnotes be printed

by the output routine, since footnotes should be printed per HTML page instead of per PDF page.

See section 60.4 for the implementation.

60.2 Minipage footnotes

See section 60.5 for how minipage footnotes are gathered. See section 94.4 for how minipage footnotes are placed into the document.

60.3 Titlepage thanks

See section 69.7 for titlepage footnotes.

60.4 Regular page footnote implementation

for HTML & PRINT:
```
\begin{warpall}
```

`FootnoteDepth` (*Ctr*)
Default: 3

Determines how deeply to place footnotes in the HTML files, similar to `tocdepth`. The default of 3 places footnotes before each `\subsubsection` or higher. See table 12 for a table of LaTeX section headings.

```
\newcounter{FootnoteDepth}
\setcounter{FootnoteDepth}{3}
```

`footnoteReset` (*Ctr*)
Default: 0

If non-zero, the `footnote` counter is reset to this value each time the footnotes are printed, as controlled by `FootnoteDepth`. For the manyfoot and bigfoot packages, additional counters such as `footnote<suffix>Reset` will be defined as well. These counters may be set non-zero by the user, and are also set if the perpage's `\MakePerPage` or `\MakeSortedPerPage` macros are used for the `footnote` or `footnote<suffix>` counters.

(The name is not capitalized because it is made from the counter's name with "Reset" appended.)

```
\newcounter{footnoteReset}
\setcounter{footnoteReset}{0}
```

```
\end{warpall}
```

for HTML output:
```
\begin{warpHTML}
```

Required for footnotes inside `description` or amstheorem square braces:

```
\AtBeginDocument{
\robustify{\footnote}
\robustify{\footnotemark}
}
```

\LWR@footnotebox Patch LaTeX footnotes to use a new \box instead of an insert for lwarp footnotes. This avoids having the original \footins appear at the bottom of a lateximage, which is on its own new page.

```
\newbox\LWR@footnotebox
```

LWR@spewingnotes (*bool*) Used with the footnote package to suppress paragraph tags before and after \spewnotes.

```
\newbool{LWR@spewingnotes}% For the footnote package.
```

Much of the following has unneeded print-mode formatting removed.

\@makefntext {⟨*text*⟩}

```
\long\def\@makefntext#1{\textsuperscript{\@thefnmark}~{#1}}
```

\@makefnmark

```
\def\@makefnmark{%
    \textsuperscript{\@thefnmark}%
}
```

Footnotes may be in regular text, in which case paragraphs are tagged, or in a table data cell or lateximage, in which case paragraph tags must be added manually.

In a lateximage during HTML output, the lateximage is placed inside a print-mode minipage, but the footnotes are broken out by:

```
\def\@mpfn{footnote}
\def\thempfn{\thefootnote}
\let\@footnotetext\LWR@footnotetext
```

\LWR@@footnotetext {⟨*text*⟩} {⟨*footnote box name*⟩}

Factored to allow multiple footnote boxes for manyfoot.

```
\long\def\LWR@@footnotetext#1#2{%
\LWR@traceinfo{LWR@footnotetext}%
```

Perhaps generate an autopage in the text to link a citation backreference closer to its usage.

```
\LWR@newautopagelabel{page}%
\LWR@ensuredoingapar%
```

Locally disable auto page labels inside the footnote text. Footnotes are accumulated in the current page before finally being placed in a potentially later page, so the autopages would be incorrect.

```
\begingroup%
\let\LWR@newautopagelabel\LWR@null@newautopagelabel%
```

Take the existing footnote box and add the new content:

```
\global\setbox\csname #2\endcsname=\vbox{%
    \unvbox\csname #2\endcsname%
```

Remember the footnote number for `\ref`:

```
    \protected@edef\@currentlabel{%
        \csname p@footnote\endcsname\@thefnmark%
    }% @currentlabel
```

Open a group:

```
    \color@begingroup%
```

Disable CJK xpinyin while generating footnotes.

```
    \LWR@disablepinyin%
```

Use HTML superscripts in the footnote even when the main text is inside a `lateximage`, because the footnote will be in HTML:

```
    \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a `lateximage`:

```
    \ifbool{LWR@spewingnotes}{}{%
        \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
    }%
```

Append the footnote to the list:

```
    \@makefntext{#1}%
```

Closing paragraph tag:

```
    \ifbool{LWR@spewingnotes}{}{%
        \LWR@origtilde\LWR@orignewline%
        \LWR@htmltagc{/\LWR@tagregularparagraph}%
        \LWR@orignewline%
    }%
```

Close the group:

```
    \color@endgroup%
}% vbox
\endgroup%
}%
```

`\LWR@footnotetext {`⟨*text*⟩`}`

```
\long\def\LWR@footnotetext#1{\LWR@@footnotetext{#1}{LWR@footnotebox}}%
```

`\@footnotetext {`⟨*text*⟩`}`

```
\LetLtxMacro\@footnotetext\LWR@footnotetext
```

60.5 Minipage footnote implementation

Patch LaTeX minipage footnotes to use a new \box instead of an insert for lwarp minipage footnotes. This avoids having the original \@mpfootins appear at the bottom of a lateximage, which is on its own new page.

```
\newbox\LWR@mpfootnotes
```

\@mpfootnotetext {⟨*text*⟩}

```
\long\def\@mpfootnotetext#1{%
\LWR@traceinfo{@mpfootnotetext}%
\LWR@ensuredoingapar%
\global\setbox\LWR@mpfootnotes\vbox{%
    \unvbox\LWR@mpfootnotes%
    \reset@font\footnotesize%
    \hsize\columnwidth%
    \@parboxrestore%
    \protected@edef\@currentlabel%
        {\csname p@mpfootnote\endcsname\@thefnmark}%
    \color@begingroup%
```

Add paragraph tag:

```
    \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%

    \@makefntext{%
        \ignorespaces#1%
    }%
```

Add the closing paragraph tag:

```
    \leavevmode\LWR@orignewline%
    \LWR@htmltagc{/\LWR@tagregularparagraph}%

    \color@endgroup%
}% vbox
```

Paragraph handling:

```
\LWR@ensuredoingapar%
\LWR@traceinfo{@mpfootnotetext: done}%
}
```

\thempfootnote Redefined to remove the \itshape, which caused an obscure compiling error in some situations.

```
\AtBeginDocument{
    \def\thempfootnote{\@alph\c@mpfootnote}
}
```

60.6 Printing pending footnotes

\LWR@@printpendingfootnotes {⟨*footnote counter name*⟩}

```
\newcommand*{\LWR@@printpendingfootnotes}[1]{%
\expandafter\ifvoid\csname LWR@#1box\endcsname\else
    \LWR@forcenewpage
    \begin{BlockClass}(note){footnotes}%
```

Create a new autopage in case citation back references occur inside the footnotes:

```
    \LWR@newautopagelabel{page}%

    \null
    \unvbox\csuse{LWR@#1box}
    \setbox\csuse{LWR@#1box}=\vbox{}
    \end{BlockClass}
    \ifltxcounter{#1Reset}{%
        \ifnumgreater{\value{#1Reset}}{0}{%
            \setcounter{#1}{\value{#1Reset}}%
            \addtocounter{#1}{-1}%
        }{}%
    }{}%
\fi
}
```

`\LWR@printpendingfootnotes` Enclose the footnotes in a class, print, then clear. For manynotes, new footnotes may be added via `\appto`.

```
\newcommand*{\LWR@printpendingfootnotes}{%
    \LWR@@printpendingfootnotes{footnote}%
}
```

`\LWR@maybeprintpendingfootnotes` {⟨*depth*⟩} Used to print footnotes before sections only if formatting for an EPUB or word processor:

```
\newcommand*{\LWR@maybeprintpendingfootnotes}[1]{%
\ifboolexpr{
    not test{\ifnumcomp{#1}{>}{\value{FootnoteDepth}}} or
    bool{FormatEPUB} or
    bool{FormatWP}
}%
{\LWR@printpendingfootnotes}%
{}%
}
```

`\LWR@printpendingmpfootnotes` Enclose the minipage footnotes in a class, print, then clear.

```
\newcommand*{\LWR@printpendingmpfootnotes}{%
\ifvoid\LWR@mpfootnotes\else
    \LWR@forcenewpage
    \begin{BlockClass}(note){footnotes}%
    \null
    \unvbox\LWR@mpfootnotes
    \setbox\LWR@mpfootnotes=\vbox{}
    \end{BlockClass}
\fi
}
```

`\LWR@nullifyfootnotes` Cancels footnotes, such as inside an HTML comment or a `\nameref`.

```
\newcommand*{\LWR@nullifyfootnotes}{%
    \renewcommand{\footnote}[2][]{}%
    \renewcommand{\footnotemark}[1][]{}%
}
```

```
\end{warpHTML}
```

61 Marginpars

`\marginpar` `[`⟨*left*⟩`]` `{`⟨*right*⟩`}` `\marginpar` may contains paragraphs, but in order to remain inline with the surrounding text lwarp nullifies block-related macros inside the `\marginpar`. Paragraph breaks are converted to `<br />` tags.

`\marginparBlock` `[`⟨*left*⟩`]` `{`⟨*right*⟩`}` To include block-related macros, use `\marginparBlock`, which takes the same arguments but creates a `<div>` instead of a `<span>`. A line break will occur in the text where the `\marginBlock` occurs.

for HTML output:

```
\begin{warpHTML}
```

`\marginpar` `[`⟨*left*⟩`]` `{`⟨*right*⟩`}`

```
\renewcommand{\marginpar}[2][]{%
\ifbool{FormatWP}%
{%
    \begin{LWR@BlockClassWP}%
        {width:2in; float:right; margin:10pt}{}(note){marginblock}%
    #2
    \end{LWR@BlockClassWP}%
}%
{%
    \LWR@htmlspanclass(note){marginpar}{#2}%
}%
}
```

`\marginparBlock` `[`⟨*left*⟩`]` `{`⟨*right*⟩`}`

For use when the marginpar will be more than one paragraph, and/or contains more than simple text.

HTML version.

```
\newcommand{\marginparBlock}[2][]{%
    \LWR@stoppars%
    \ifbool{FormatWP}%
    {%
        \begin{LWR@BlockClassWP}%
            {width:2in; float:right; margin:10pt}{}%
            (note){marginblock}%
        #2
        \end{LWR@BlockClassWP}
    }%
    {%
        \begin{BlockClass}[width:2in; float:right; margin:10pt]%
            (note){marginparblock}%
        #2
```

```
        \end{BlockClass}
    }%
    \LWR@startpars%
}
```

\reversemarginpar

```
\renewcommand*{\reversemarginpar}{}
```

\normalmarginpar

```
\renewcommand*{\normalmarginpar}{}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

\marginparBlock [⟨*left*⟩] {⟨*right*⟩}

For use when the marginpar will be more than one paragraph, and/or contains more than simple text.

Print version.

```
\LetLtxMacro\marginparBlock\marginpar
```

```
\end{warpprint}
```

62 Tracking internal cross references

Cross references are generated using the PDF file's page number during LaTeX compilation. Internal labels are generated which include these page numbers in the label.

*_html.aux (*file*) A new entry in the *_html.aux file is used to help cross-references:

```
\newlabel{autopage-<nnn>}{{<x>}{<y>}}
```

LWR@currentautosecpage (*Ctr*) Records the page number when the section was created. (If a math expression is included in the section name, and SVG math is used, the corresponding lateximage will cause the page number to change by the time the following autosec label is created, thus the initial page number is recorded here.) LWR@currentautosecfloatpage is updated more often than LWR@currentautosecpage.

```
\newcounter{LWR@currentautosecpage}
\setcounter{LWR@currentautosecpage}{1}
```

LWR@currentautosecfloatpage (*Ctr*) The HTML output's PDF page number at the start of a new HTML file, section, or float. Updated more often than LWR@currentautosecpage, such as when a new float occurs. Used only for table of contents, list of figures, list of tables, but not for general cross references such as \label, citation backlinks, etc.

\LWRsetnextfloat is written with this and the autoid by the modified \addcontentsline just before each float's entry.

```
\newcounter{LWR@currentautosecfloatpage}
\setcounter{LWR@currentautosecfloatpage}{1}
```

LWR@previousautopagelabel (*Ctr*) Remembers which autopage label was most recently generated. Used to avoid duplicates.

```
\newcounter{LWR@previousautopagelabel}
\setcounter{LWR@previousautopagelabel}{-1}
```

\LWR@newautopagelabel {⟨*pagenumber counter*⟩}

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\newcommand*{\LWR@newautopagelabel}[1]{%
```

No action if this autopage label has already been defined:

```
\ifnumequal{\value{LWR@previousautopagelabel}}{\value{page}}%
    {}%
```

If the PDF page has changed, create a label using the desired counter.

If the counter is LWR@currentautosecpage, that was the page number when the section generation began, but the current PDF page may be different by now if the section name had an SVG image, such as SVG math. To allow the cross-reference to point just after the section heading, the label must be made after the section heading is complete, which may have generated a new PDF page. Thus, the label is made with the given counter, which may be the PDF page number where the section heading began, then if the PDF page number has changed, another label is made for the current page number.

```
    {%
        \label{\BaseJobname-autopage-\csuse{the#1}}%
```

If there are intervening pages, such as an SVG image, define another label for the new page:

```
        \ifnumequal{\value{#1}}{\value{page}}%
            {}%
            {\label{\BaseJobname-autopage-\csuse{thepage}}}%
```

Remember the latest autopage label:

```
        \setcounter{LWR@previousautopagelabel}{\value{page}}%
    }%
}
```

\LWR@null@newautopagelabel {⟨*pagenumber counter*⟩}

Inside a footnote, the page numbers will be incorrect, so this is nullified.

```
\newcommand*{\LWR@null@newautopagelabel}[1]{}
```

63 Splitting HTML files

- Files are split according to `FileDepth` and `CombineHigherDepths`.
- Filenames are sanitized by `\LWR@filenamenoblanks`.
- `\LWR@newhtmlfile` finishes an HTML page, adds a comment to tell where and how to split the file, then starts a new HTML page.

for HTML & PRINT:
```
\begin{warpall}
```

`FileDepth` (*Ctr*) {⟨*section depth*⟩} determines how deeply to break into new HTML files, similar to `tocdepth`. The default of -5 produces one large HTML file.

```
\newcounter{FileDepth}
\setcounter{FileDepth}{-5}
```

`CombineHigherDepths` (*bool*) Combile higher-level sections together into one file?

```
\newbool{CombineHigherDepths}
\booltrue{CombineHigherDepths}
```

`\FilenameLimit` Maximum length of the generated filenames.

```
\newcommand*{\FilenameLimit}{80}
```

```
\end{warpall}
```

for HTML output:
```
\begin{warpHTML}
```

`\LWR@thisfilename` The currently-active filename or number. At first, this is the homepage.

```
\AtBeginDocument{
\ifbool{FileSectionNames}%
    {\newcommand*{\LWR@thisfilename}{\HomeHTMLFilename}}
    {\newcommand*{\LWR@thisfilename}{0}}
}
```

`\LWR@thisnewfilename` The filename being sanitized.

```
\newcommand*{\LWR@thisnewfilename}{}
```

`\LWR@simplifyname` * {⟨*expression*⟩} Simplify `\LWR@thisnewfilename`.

If starred, detokenizes the input expression. If found, changes the expression to a single detokenized dash.

```
\NewDocumentCommand{\LWR@simplifyname}{s m}{%
\IfBooleanTF{#1}{%
    \StrSubstitute{\LWR@thisnewfilename}%
        {\detokenize{#2}}%
        {\detokenize{-}}[\LWR@thisnewfilename]%
}{%
    \StrSubstitute{\LWR@thisnewfilename}%
```

```
        {#2}%
        {\detokenize{-}}[\LWR@thisnewfilename]%
}
}
```

`\LWR@simplifycustom` User-defined filename simplifications. Redefine with `\newcommand`.

```
\newcommand*{\LWR@simplifycustom}{}
```

`\FilenameSimplify` `*` `{`⟨*phrase*⟩`}` Assign a user-defined filename simplification. Appends to `\LWR@simplifycustom`.

```
\NewDocumentCommand{\FilenameSimplify}{s m}{%
\IfBooleanTF{#1}{%
    \appto{\LWR@simplifycustom}{%
        \LWR@simplifyname*{#2}%
    }%
}{%
    \appto{\LWR@simplifycustom}{%
        \LWR@simplifyname{#2}%
    }%
}%
}
```

`\LWR@avoiddupfilenames` Instructions for how to avoid duplicate filenames. This is used in a warning in `\LWR@filenamenoblanks`, and in an error in `\LWR@newhtmlfile`.

```
\newcommand*{\LWR@avoiddupfilenames}{%
    To avoid duplicate filenames, use the optional\MessageBreak
    short Table of Contents entry:\MessageBreak
    \space\space\protect\section[Unique name, no math]{Name with math}%
        \MessageBreak
    or use \protect\texorpdfstring, from the hyperref package:\MessageBreak
    \space\space%
        \protect\section{\MessageBreak
            \space\space\space\space\protect\texorpdfstring\MessageBreak
                \space\space\space\space\space\space%
                {Name with math}{Unique name, no math}\MessageBreak
        \space\space}
}
```

`\LWR@filenamenoblanks` `{`⟨*filename*⟩`}`

Convert blanks into dashes, removes short words, store result in `\LWR@thisfilename`.

Also see `\LWR@nullfonts` for nullified macros.

```
\newcommand*{\LWR@filenamenoblanks}[1]{%
\begingroup
```

Locally temporarily disable direct-formatting commands, not used in filenames:

```
\LWR@nullfonts%
\renewcommand*{\LWR@htmltagc}[1]{}%

\edef\LWR@thisnewfilename{#1}%
```

Replaces common macros with hyphens. (\& is done by \LWR@nullfonts.)

```
\RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
\LWR@simplifyname{\_}
\LWR@simplifyname{\#}
\LWR@simplifyname{\textbackslash}
\LWR@simplifyname{\protect}
\LWR@simplifyname{\ }
\LWR@simplifyname{\textless}
\LWR@simplifyname{\textgreater}

\edef\LWR@thisnewfilename{\detokenize\expandafter{\LWR@thisnewfilename}}%
```

Warn if there is dollar math in the section name:

```
\ifbool{FileSectionNames}{%
    \IfSubStr{\LWR@thisnewfilename}{\LWRdollar}{%
        \PackageWarning{lwarp}
        {%
            This section name:\MessageBreak
            \space\space``\detokenize\expandafter{#1}''\MessageBreak
            at the line number listed below,\MessageBreak
            is using $dollar-delimited math$,
            which generates\MessageBreak
            complicated file names.  It is better to use\MessageBreak
            \space\space%
           \protect\section{Name with \protect\(parenthesis math\protect\)}%
                \MessageBreak
            The math then will be removed from the file name.\MessageBreak
            \MessageBreak
            \LWR@avoiddupfilenames%
            \MessageBreak
            This section is found before or%
        }
    }{}%
}{}

\LWR@traceinfo{LWR@filenamenoblanks edef: !\LWR@thisnewfilename!}%
\fullexpandarg%
```

Convert spaces into hyphens:

```
\LWR@simplifyname*{ }
```

Convert punctutation into hyphens:

```
\LWR@simplifyname*{*}
\LWR@simplifyname*{(}
\LWR@simplifyname*{)}
\LWR@simplifyname*{.}
\LWR@simplifyname*{!}
\LWR@simplifyname*{,}
\LWR@simplifyname*{'}
\LWR@simplifyname*{+}
\LWR@simplifyname*{/}
\LWR@simplifyname*{:}
\LWR@simplifyname*{;}
\LWR@simplifyname*{=}
```

```
\LWR@simplifyname*{?}
\LWR@simplifyname*{@}
\LWR@simplifyname*{^}
\LWR@simplifyname*{&}
\LWR@simplifyname*{"}
\LWR@simplifyname*{<}
\LWR@simplifyname*{>}

\LWR@simplifyname{\LWRbackslash}
```

Braces are removed entirely to avoid extra dashes in the result.

```
\StrSubstitute{\LWR@thisnewfilename}%
    {\LWRleftbrace}{}[\LWR@thisnewfilename]%
\StrSubstitute{\LWR@thisnewfilename}%
    {\LWRrightbrace}{}[\LWR@thisnewfilename]%

\LWR@simplifyname{\LWRpercent}
\LWR@simplifyname{\LWRdollar}

\LWR@simplifyname*{|}
\LWR@simplifyname*{^}
\LWR@simplifyname*{~}
\LWR@simplifyname*{[}
\LWR@simplifyname*{]}
\LWR@simplifyname*{‘}
```

Convert short words:

```
\LWR@simplifyname*{-s-}
\LWR@simplifyname*{-S-}
\LWR@simplifyname*{-a-}
\LWR@simplifyname*{-A-}
\LWR@simplifyname*{-an-}
\LWR@simplifyname*{-AN-}
\LWR@simplifyname*{-to-}
\LWR@simplifyname*{-TO-}
\LWR@simplifyname*{-by-}
\LWR@simplifyname*{-BY-}
\LWR@simplifyname*{-of-}
\LWR@simplifyname*{-OF-}
\LWR@simplifyname*{-and-}
\LWR@simplifyname*{-AND-}
\LWR@simplifyname*{-for-}
\LWR@simplifyname*{-FOR-}
\LWR@simplifyname*{-the-}
\LWR@simplifyname*{-THE-}
```

Convert custom words:

```
\LWR@simplifycustom%
```

If PDF LaTeX and not `utf8` encoding, don't try to convert emdash, endash:

```
\ifPDFTeX% pdflatex or dvi latex
\ifdefstring{\inputencodingname}{utf8}{%
    \LWR@simplifyname*{—}
```

```
%     emdash
    \LWR@simplifyname*{-}
%     endash
}{}%
\else% not PDFTeX
    \LWR@simplifyname*{-}
    \LWR@simplifyname*{-}
\fi%
```

Convert multiple hyphens:

```
\LWR@simplifyname*{-----}
\LWR@simplifyname*{----}
\LWR@simplifyname*{---}
\LWR@simplifyname*{--}
```

If starts with a dash, remove the leading dash:

```
\IfBeginWith{\LWR@thisnewfilename}{\detokenize{-}}{%
    \StrGobbleLeft{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
}{}%
```

If ends with a dash, remove the trailing dash:

```
\IfEndWith{\LWR@thisnewfilename}{\detokenize{-}}{%
    \StrGobbleRight{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
}{}%
```

Limits the length of the filename:

```
\StrLeft{\LWR@thisnewfilename}{\FilenameLimit}[\LWR@thisnewfilename]%
```

Return the global result:

```
\global\let\LWR@thisfilename\LWR@thisnewfilename%
\endgroup%
\LWR@traceinfo{LWR@filenamenoblanks: result is \LWR@thisfilename}%
}
```

63.1 Sanitizing expressions for HTML

Math expressions are converted to `latexmages`, and some math environments may contain &, <, or >, which should not be allowed inside an HTML <alt> tag, so must convert them to HTML entities.

`\LWR@replacestrings` {⟨*search*⟩} {⟨*replace*⟩}

Replaces strings inside `\tmpb`.

Modified from the original, by PETR OLSAK, from the opmac package.

```
\bgroup
\catcode`!=3 \catcode`?=3

\long\gdef\LWR@replacestrings@addto#1#2{%
    \expandafter\def\expandafter#1\expandafter{#1#2}%
```

```
}

\gdef\LWR@replacestrings#1#2{%
    \long\def\LWR@replacestringsA##1#1{\def\tmpb{##1}\LWR@replacestringsB}%
   \long\def\LWR@replacestringsB##1#1{%
        \ifx!##1\relax \else\LWR@replacestrings@addto\tmpb{#2##1}%
        \expandafter\LWR@replacestringsB\fi%
    }%                                  improved version <May 2016> inspired
   \expandafter\LWR@replacestringsA\tmpb?#1!#1% from pysyntax.tex by Petr Krajnik
    \long\def\LWR@replacestringsA##1?{%
        \def\tmpb{##1}%
    }\expandafter\LWR@replacestringsA\tmpb%
}
\egroup
```

LWR@MathJax@silentquotes (*bool*)

If true, double quotes (\" and ") are removed (used for mathspec). This unfortunately includes double quotes used inside \text with MathJax. If false, double quotes are escaped.

```
\newbool{LWR@MathJax@silentquotes}
\boolfalse{LWR@MathJax@silentquotes}
```

\LWR@subHTMLsanitize

\LWR@strresult must first be set by \LWR@HTMLsanitize, \LWR@HTMLsanitizeexpand, or \CustomizeMathJax.

```
\catcode`\#=12
\catcode`\&=12
\newcommand{\LWR@subHTMLsanitize}{%
```

The &, <, and > may be interpreted by the browser:

```
    \edef\tmpb{\detokenize\expandafter{\LWR@strresult}}%
    \LWR@replacestrings{&}{&amp;}%
    \LWR@replacestrings{<}{&lt;}%
    \LWR@replacestrings{>}{&gt;}%
```

The quotes occasionally causes problems. For mathspec, also allow neutralization of \" and the " character.

```
    \ifbool{LWR@MathJax@silentquotes}
        {%
            \expandafter\LWR@replacestrings\expandafter{\LWRbackslash"}{}%
            \LWR@replacestrings{"}{}%
        }%
        {\LWR@replacestrings{"}{&quot;}}%
    \LWR@replacestrings{'}{&apos;}%
    \LWR@replacestrings{`}{&grave;}%
```

MathJax allows expressions to be defined with \newcommand. These expressions would appear with ## for each argument, and each must be changed to a single #. This must be done after all the above changes. Attempting another conversion after this causes an error upon further expansion.

```
    \LWR@replacestrings{##}{#}%
    \edef\LWR@strresult{\detokenize\expandafter{\tmpb}}%
}
```

```
\catcode`\#=6
\catcode`\&=4
```

`\LWR@HTMLsanitizedetokenized` {⟨*detokenized text*⟩}

Prints the sanitized text, already detokenized.

```
\newrobustcmd{\LWR@HTMLsanitizedetokenized}[1]{%
    \LWR@traceinfo{LWR@HTMLsanitizedetokenized !#1!}%
```

Cancel French babel character handling, and fully expand the strings:

```
    \begingroup%
    \LWR@hook@processingtags%
    \edef\LWR@strresult{#1}%
    \LWR@subHTMLsanitize%
    \LWR@strresult%
    \endgroup%
    \LWR@traceinfo{LWR@HTMLsanitize done}%
}
```

`\LWR@HTMLsanitizeexpanded` {⟨*text*⟩}

This version must be given the detokenized and expanded text. This is only used for adding math to MathJax expressions or `lateximage` `alt` tags.

```
\edef\LWR@beginspaceleftbrace{begin \LWRleftbrace}
\edef\LWR@beginspaceleftbrace{\detokenize\expandafter{\LWR@beginspaceleftbrace}}
\edef\LWR@beginleftbrace{begin\LWRleftbrace}
\edef\LWR@beginleftbrace{\detokenize\expandafter{\LWR@beginleftbrace}}

\edef\LWR@endspacerightbrace{end \LWRrightbrace}
\edef\LWR@endspacerightbrace{\detokenize\expandafter{\LWR@endspacerightbrace}}
\edef\LWR@endrightbrace{end\LWRrightbrace}
\edef\LWR@endrightbrace{\detokenize\expandafter{\LWR@endrightbrace}}

\newrobustcmd{\LWR@HTMLsanitizeexpanded}[1]{%
```

Cancel French babel character handling, and fully expand the strings:

```
    \begingroup%
    \LWR@hook@processingtags%

    \edef\LWR@strresult{#1}%
```

The math expression may includes spaces between tokens, but MathJax does not want a space between `\begin` or `\end` and the following brace. This space is removed here.

```
    \protect\StrSubstitute{\LWR@strresult}%
        {\LWR@beginspaceleftbrace}{\LWR@beginleftbrace}[\LWR@strresult]%
    \protect\StrSubstitute{\LWR@strresult}%
        {\LWR@endspacerightbrace}{\LWR@endrightbrace}[\LWR@strresult]%

    \LWR@subHTMLsanitize%
    \LWR@strresult%
    \endgroup%
}
```

63.2 Customizing MathJax

\LWR@customizedMathJax Additional MathJax definitions to be added to the start of each HTML page.

```
\newcommand*{\LWR@customizedMathJax}{}
```

LWR@warnedcustomizemathjax (*bool*) Used to issue only one warning about using a \CustomizeMathJax per macro.

```
\newbool{LWR@warnedcustomizemathjax}
\boolfalse{LWR@warnedcustomizemathjax}
```

\LWR@subcustomizedmathjax {⟨*macro definition*⟩}

```
\newcommand*{\LWR@subcustomizedmathjax}[1]{%
    \begingroup%
    \LWR@hook@processingtags%
    \edef\LWR@strresult{\detokenize{#1}}%
    \LWR@subHTMLsanitize%
    \xdef\LWR@customizedMathJax{%
        \LWR@customizedMathJax%
            \LWR@strresult%
    }%
    \endgroup%
}
\@onlypreamble\LWR@subcustomizedmathjax
```

\CustomizeMathJax {⟨*macro definition*⟩}

A warning is issued if a very long argument is given.

```
\newcommand*{\CustomizeMathJax}[1]{%
    \ifbool{LWR@warnedcustomizemathjax}{}{%
        \StrLen{\detokenize{#1}}[\LWR@tempone]%
        \ifnumgreater{\LWR@tempone}{350}{%
            \AtEndDocument{%
                \PackageNoteNoLine{lwarp}{%
                To ensure faster MathJax compilation, place each\MessageBreak
              custom macro in its own \protect\CustomizeMathJax.\MessageBreak
                See the Lwarp documentation regarding customizing\MessageBreak
                     MathJax%
                }%
            }%
            \booltrue{LWR@warnedcustomizemathjax}%
        }{}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \LWR@subcustomizedmathjax{#1}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
}
\@onlypreamble\CustomizeMathJax
```

\LWR@infoprocessingmathjax {⟨*package name*⟩}

```
\newcommand*{\LWR@infoprocessingmathjax}[1]{%
\typeout{---}
\typeout{Package lwarp: Processing MathJax customizations for #1.}
```

```
\typeout{\space\space This may take a moment.}
\typeout{---}
}
```

defaults Default customizations:

In the MathJax code, footnotes are only referenced. For equations, they are also generated in the HTML when the LaTeX math is generated inside the HTML comment. For other math environments, the `\footnotemark`/`\footnotetext` method must be used. See section 8.5.4 regarding `\footnotemark`.

⚠ `\footnotemark`

For footnotes, `\footnotename` is used in most cases, however for `equation` the footnote is picked up from LaTeXin `\LWR@doendequation`.

First, `\footnotename` for MathJax is copied from LaTeX.

```
\providecommand{\footnotename}{footnote}

% due to warpMathJax:
\end{warpHTML}

\begin{warpMathJax}
\xdef\LWR@customizedMathJax{\LWR@customizedMathJax%
    \LWRbackslash(%
    \LWRbackslash{}newcommand%
    \{\LWRbackslash{}footnotename\}%
    \{\footnotename\}%
    \LWRbackslash)\par%
}
\end{warpMathJax}
```

`\LWRfootnote` is set per `equation` if a footnote is detected in the equation's math expression, otherwise it defaults to `\footnotename`.

```
\begin{warpMathJax}
\CustomizeMathJax{\def\LWRfootnote{1}}
\CustomizeMathJax{\newcommand{\footnote}[2][\LWRfootnote]{{}^{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\footnotemark}[1][\LWRfootnote]{{}^{\mathrm{#1}}}}
```

`\hspace` is modified to accept and ignore a star:

```
\CustomizeMathJax{\let\LWRorighspace\hspace}
\CustomizeMathJax{\renewcommand{\hspace}{\ifstar\LWRorighspace\LWRorighspace}}
```

Various other customizations:

```
\CustomizeMathJax{\newcommand{\mathnormal}[1]{{#1}}}
\CustomizeMathJax{\newcommand\ensuremath[1]{#1}}
\CustomizeMathJax{% absorb two optional arguments
    \newcommand{\LWRframebox}[2][]{\fbox{#2}}
    \newcommand{\framebox}[1][]{\LWRframebox}
}
\CustomizeMathJax{\newcommand{\setlength}[2]{}}
\CustomizeMathJax{\newcommand{\addtolength}[2]{}}
\CustomizeMathJax{\newcommand{\setcounter}[2]{}}
\CustomizeMathJax{\newcommand{\addtocounter}[2]{}}
\CustomizeMathJax{\newcommand{\arabic}[1]{}}
\CustomizeMathJax{\newcommand{\number}[1]{}}
```

```
\CustomizeMathJax{\newcommand{\noalign}[1]{\text{#1}\notag \\}}
\CustomizeMathJax{\newcommand{\cline}[1]{}}
\CustomizeMathJax{\newcommand{\directlua}[1]{\text{(directlua)}}}
\CustomizeMathJax{\newcommand{\luatexdirectlua}[1]{\text{(directlua)}}}
```

\protect, \mathchar, and \delimiter are silently discarded; and \mathcode and \delcode are ignored.

```
\CustomizeMathJax{\newcommand{\protect}{}}
\CustomizeMathJax{\def\LWRabsorbnumber#1 {}}
\CustomizeMathJax{\def\LWRabsorbquotenumber"#1 {}}
\CustomizeMathJax{\newcommand{\LWRabsorboption}[1][]{}}
\CustomizeMathJax{\newcommand{\LWRabsorbtwooptions}[1][]{\LWRabsorboption}}
\CustomizeMathJax{\def\mathchar{\ifnextchar"\LWRabsorbquotenumber\LWRabsorbnumber}}
\CustomizeMathJax{\def\mathcode#1={\mathchar}}
\CustomizeMathJax{\let\delcode\mathcode}
\CustomizeMathJax{\let\delimiter\mathchar}
```

Some text symbols missing from MathJax:

```
\CustomizeMathJax{\def\oe{\unicode{x0153}}}
\CustomizeMathJax{\def\OE{\unicode{x0152}}}
\CustomizeMathJax{\def\ae{\unicode{x00E6}}}
\CustomizeMathJax{\def\AE{\unicode{x00C6}}}
\CustomizeMathJax{\def\aa{\unicode{x00E5}}}
\CustomizeMathJax{\def\AA{\unicode{x00C5}}}
\CustomizeMathJax{\def\o{\unicode{x00F8}}}
\CustomizeMathJax{\def\O{\unicode{x00D8}}}
\CustomizeMathJax{\def\l{\unicode{x0142}}}
\CustomizeMathJax{\def\L{\unicode{x0141}}}
\CustomizeMathJax{\def\ss{\unicode{x00DF}}}
\CustomizeMathJax{\def\SS{\unicode{x1E9E}}}
\CustomizeMathJax{\def\dag{\unicode{x2020}}}
\CustomizeMathJax{\def\ddag{\unicode{x2021}}}
\CustomizeMathJax{\def\P{\unicode{x00B6}}}
\CustomizeMathJax{\def\copyright{\unicode{x00A9}}}
\CustomizeMathJax{\def\pounds{\unicode{x00A3}}}
\end{warpMathJax}


\begin{warpHTML}% due to warpMathJax
```

`\LWR@customizeMathJax` Prints MathJax commands to the HTML output.

```
\newcommand{\LWR@customizeMathJax}{%
\ifbool{mathjax}{

\LWR@stoppars
\LWR@htmlcomment{MathJax customizations:}

\begin{BlockClass}{hidden}
\LWR@stoppars
```

Avoid ligatures while printing MathJax customizations:

```
{
    \LWR@print@ttfamily
    \LWR@customizedMathJax
```

```
}
\LWR@startpars
\end{BlockClass}

\LWR@startpars
}{}
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

`\CustomizeMathJax` The print-mode version:

```
\newcommand*{\CustomizeMathJax}[1]{}
```

`\FilenameSimplify` `*` {⟨*expression*⟩}

```
\NewDocumentCommand{\FilenameSimplify}{s m}{}
```

```
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@createfooter` If specified, create the first or later web page footer.

```
\newcommand*{\LWR@createfooter}{%
    \ifnumless{\value{LWR@htmlseqfilenumber}}{1}{%
        \ifdefempty{\LWR@firstpagebottom}{}{%
            \LWR@htmlelement{footer}

            \LWR@firstpagebottom

            \LWR@htmlelementend{footer}
        }%
    }{%
        \ifdefempty{\LWR@pagebottom}{}{%
            \LWR@htmlelement{footer}

            \LWR@pagebottom

            \LWR@htmlelementend{footer}
        }%
    }%
}
```

`\LWR@newhtmlfile` {⟨*section name*⟩}

Finishes the current HTML page with footnotes, footer, navigation, then starts a new HTML page with an HTML comment telling where to split the page and what the new filename and CSS are, then adds navigation, side TOC, header, and starts the text body.

```
\newcommand*{\LWR@newhtmlfile}[1]{
\LWR@traceinfo{LWR@newhtmlfile}
```

At the bottom of the ending file:

```
\LWR@htmlelementclassend{section}{textbody}
\LWR@htmlelementclassend{main}{bodycontainer}
\LWR@htmlelementclassend{div}{bodyandsidetoc}

\LWR@printpendingfootnotes

```

No footer between files if EPUB:

```
\ifbool{FormatEPUB}{}{\LWR@createfooter}
```

No bottom navigation if are finishing the home page or formatting for EPUB or a word-processor.

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {\ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}}
```

End of this HTML file:

```
\LWR@stoppars
\LWR@htmltag{/body}\LWR@orignewline
\LWR@htmltag{/html}\LWR@orignewline
\LWR@traceinfo{LWR@newhtmlfile: about to LWR@orignewpage}
\LWR@maybe@orignewpage


\addtocounter{LWR@htmlfilenumber}{1}%
\addtocounter{LWR@htmlseqfilenumber}{1}%
```

If using a filename based on section name, create a version without blanks. The filename without blanks will be placed into `\LWR@thisfilename`. Duplicates will be detected using MD5 hashes.

If not using a filename, the file number will be used instead.

```
\ifbool{FileSectionNames}%
{%
```

Convert the section name to a filename with blanks and common words removed. The resulting filename is in `\LWR@thisfilename`.

```
    \LWR@filenamenoblanks{#1}%
```

Create a macro name from the MD5 hash of the file name, to detect duplicates:

```
    \edef\LWR@hashedname{\LWR@mdfive{\LWR@thisfilename}}%
```

If the macro name is not yet defined, this filename is unique.

```
    \ifcsundef{LWR@filename\LWR@hashedname}{%
```

If the filename is unique, create a macro using the hashed name, to be used to test for additional duplicates in the future.

```
        \csdef{LWR@filename\LWR@hashedname}{}%
    }{%
```

If the filename is not unique, create an error.

```
        \PackageError{lwarp}%
            {%
                The section name:\MessageBreak
                ``#1'',\MessageBreak
                at the line number listed below,\MessageBreak
                generates the filename\MessageBreak
                ``\LWR@thisfilename'',\MessageBreak
                which appears to be a duplicate.  There is a\MessageBreak
             previous section with an identical or similar name.\MessageBreak
              While generating file names, Lwarp sanitizes math,\MessageBreak
                most symbols, and a few common short words,\MessageBreak
                and this may cause a conflict.\MessageBreak
                Enter 'H' for possible solutions%
            }%
            {%
                \LWR@avoiddupfilenames%
            }%
    }%
}%
```

If using file numbers instead of names, the name is set to the next file number.

```
{\renewcommand*{\LWR@thisfilename}{\arabic{LWR@htmlfilenumber}}}
```

Include an HTML comment to instruct lwarpmk where to split the files apart. Uses pipe-separated fields for `split_html.gawk`. Uses monospaced font with ligatures disabled for everything except the title.

```
\LWR@traceinfo{LWR@newhtmlfile: about to print start file}%
```

`\LWR@nullfonts` to allow math in a section name.

```
\begingroup%
\LWR@nullfonts%
\LWR@htmlblockcomment{%
|Start file|%
\LWR@htmlsectionfilename{\LWR@thisfilename}|%
}
\endgroup%
```

At the top of the starting file:

```
\LWR@stoppars

```

Start a new file with the given section name:

```
\LWR@filestart[#1]

```

Track the PDF page numbers of the HTML output. This is updated more frequently than `LWR@currentautosecpage`.

```
\setcounter{LWR@currentautosecfloatpage}{\value{page}}%
\LWR@newautopagelabel{LWR@currentautosecfloatpage}%
```

No navigation between files if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {\LWR@topnavigation}

```

No header if between files if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {


        \ifdefempty{\LWR@pagetop}{}{
            \LWR@htmlelement{header}

            \LWR@pagetop

            \LWR@htmlelementend{header}
        }
    }

```

The container for the sidetoc and text body:

```
\LWR@htmlelementclass{div}{bodyandsidetoc}
```

No sideTOC if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {\LWR@sidetoc}

```

Start of the <textbody>:

```
\LWR@htmlelementclass{main}{bodycontainer}
\LWR@htmlelementclass{section}{textbody}
```

Not yet found a new section in this file. Once one is found, a label will be placed for previous/next links.

```
\boolfalse{LWR@setseqfilelabel}
```

Print title only if there is one. Skip if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}%
    {}%
    {%
        \ifcsvoid{thetitle}{}{%
            \LWR@printthetitle%
        }%
    }%
```

Keep paragraph tags disabled for now:

```
\LWR@stoppars
```

If using MathJax, print the customizations here.

```
\LWR@customizeMathJax

\LWR@traceinfo{LWR@newhtmlfile: done}
}

\end{warpHTML}
```

64 Sectioning

Sectioning and cross-references have been emulated from scratch, rather than try to patch several layers of existing LaTeX code and packages. Formatting is handled by css, so the emulated code has much less work to do than the print versions.

Unicode

⚠ accents in filenames

Section names and the resulting filenames with accented characters are partially supported, depending on the ability of *pdflatex* to generate characters and *pdftotext* to read them. If extra symbols appear in the text, it may be that *pdflatex* is actually producing a symbol over or under a character, resulting in *pdftotext* picking up the accent symbol separately.

XeLaTeX and LuaLaTeX directly support accented section and file names, but it may be necessary to use LaTeX accents instead of native Unicode accents. LaTeX accents will have the accents stripped when creating file names, whereas using Unicode accents will create filenames which include accents, which may cause issues with some operating systems.

for HTML output:

```
\begin{warpHTML}
```

64.1 User-level starred section commands

\ForceHTMLPage

For HTML output, forces the next section to be on its own HTML page, if `FileDepth` allows, even if starred. For use with `\printindex` and others which generate a starred section which should be on its own HTML page. Also see `\ForceHTMLTOC`.

For print output, no effect.

```
\newbool{LWR@forcinghtmlpage}
\boolfalse{LWR@forcinghtmlpage}

\newcommand*{\ForceHTMLPage}{%
\global\booltrue{LWR@forcinghtmlpage}%
}
```

\ForceHTMLTOC

For HTML output, forces the next section to have a TOC entry, even if starred. For use with `\printindex` and others which generate a starred section which should be in the TOC so that it may be accessed via HTML. Not necessary if used with tocbibind. Also see `\ForceHTMLPage`.

For print output, no effect.

```
\newbool{LWR@forcinghtmltoc}
\boolfalse{LWR@forcinghtmltoc}

\newcommand*{\ForceHTMLTOC}{%
\global\booltrue{LWR@forcinghtmltoc}%
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\ForceHTMLPage}{}
\newcommand*{\ForceHTMLTOC}{}
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

64.2 Book class commands

`\mainmatter` Declare the main matter section of the document. Does not reset the page number, which must be consecutive arabic numbers for the HTML conversion.


```
\newbool{LWR@mainmatter}
\DeclareDocumentCommand{\mainmatter}{}{%
\booltrue{LWR@mainmatter}%
}
```

`\frontmatter` Declare the front matter section of the document, using arabic numbering for the internal numbering. Does not reset the page number.

```
\DeclareDocumentCommand{\frontmatter}{}{%
\boolfalse{LWR@mainmatter}%
}
```

`\backmatter` Declare the back matter section of the document. Does not reset the page number.

```
\DeclareDocumentCommand{\backmatter}{}{%
\boolfalse{LWR@mainmatter}
}
```

64.3 Sectioning support macros

`\LWR@sectionnumber` {⟨*section type*⟩}

Typeset a section number and its trailing space with CSS formatting:

```
\newcommand*{\LWR@sectionnumber}[1]{%
\InlineClass{sectionnumber}{#1}%
}
```

`autosec` A tag used by the TOC and index.

`\LWR@createautosec` {⟨*section type*⟩}

Create an autosection tag.

The use of `\textquotedbl` instead of " provides improved compatibility with xeCJK.

```
\newcommand*{\LWR@createautosec}[1]{%
\LWR@htmltag{%
    #1 % space
    id=\textquotedbl\LWR@print@mbox{autosec-\arabic{page}}\textquotedbl%
}%
}
```

`\LWR@pushoneclose` {⟨*sectiontype*⟩} Stacks the new sectioning level's closing tag, to be used when this section is closed some time later.

⚠ `\LWR@stoppars` must be executed first.

```
\NewDocumentCommand{\LWR@pushoneclose}{m}{%
\LWR@traceinfo{LWR@pushoneclose #1}%
    \LWR@pushclose{#1}%
}
```

`\LWR@startnewdepth` {⟨*sectiontype*⟩}

Closes currently stacked tags of a lesser level, then opens the new nesting level by saving this new sectioning level's closing tag for later use.

⚠ `\LWR@stoppars` must be executed first.

```
\NewDocumentCommand{\LWR@startnewdepth}{m}{%
```

Close any stacked sections up to this new one.

```
\LWR@closeprevious{#1}%
```

Push a new section depth:

```
\LWR@pushoneclose{#1}%
}
```

`LWR@prevFileDepth` (*Ctr*) Remembers the previous `LWR@FileDepth`.

Initialized to a deep level so that any section will trigger a new HTML page after the home page.

```
\newcounter{LWR@prevFileDepth}
\setcounter{LWR@prevFileDepth}{\LWR@depthsubparagraph}
```

`\@seccntformat` {⟨*sectiontype*⟩}

```
\def\@seccntformat#1{\csname the#1\endcsname\quad}
```

`\simplechapterdelim` Used by tocbibind and anonchap.

```
\newcommand*{\simplechapterdelim}{}
```

`\@chapcntformat` {⟨*sectiontype*⟩}

`\let` to `\@seccntformat` by default, but may be redefined by `\simplechapter` and `\restorechapter` from tocbibind or anonchap.

```
\let\@chapcntformat\@seccntformat
```

`\@partcntformat` {⟨*sectiontype*⟩}

`\let` to `\@seccntformat` by default, but may be redefined by ctex.

```
\let\@partcntformat\@seccntformat
```

`\@partnameformat` Prints "Part" for part sections.

Nullified by ctex.

```
\newcommand*{\@partnameformat}{\LWR@isolate{\partname}~}%
```

`\LWR@printchaptername` Print `\chaptername` in most cases, but this is nullified for ctexbook, komascript, ujt* classes.

```
\newcommand*{\LWR@printchaptername}{%
    \ifdefvoid{\chaptername}{}{\chaptername~}%
}
```

`\LWR@section` * [⟨*TOC name*⟩] {⟨*name*⟩} {⟨*sectiontype*⟩}

The common actions for the high-level sectioning commands.

```
\DeclareDocumentCommand{\LWR@section}{m m m m}{%
\IfValueTF{#2}%
    {\LWR@traceinfo{LWR@section: starting #4 #2}}%
    {\LWR@traceinfo{LWR@section: starting #4 #3}}%
```

Warn if starting a section inside a :

```
    \LWR@spanwarninvalid{section}%
```

```
\LWR@maybeprintpendingfootnotes{\csuse{LWR@depth#4}}%
\LWR@stoppars%
\LWR@startnewdepth{#4}%
```

Cancel special `minipage` horizontal space interaction:

```
\global\boolfalse{LWR@minipagethispar}%
```

Start a new HTML file unless starred, and if is a shallow sectioning depth.

Exception: Also start a new HTML file for `\part*`, for appendix.

Generate a new LaTeX page so that TOC and index page number points to the section:

```
\LWR@traceinfo{LWR@section: testing whether to start a new HTML file}%
```

```
\IfBooleanT{#1}{\LWR@traceinfo{LWR@section: starred}}%
\ifbool{LWR@forcinghtmlpage}{\LWR@traceinfo{LWR@section: forcinghtmlpage}}{}%
\ifthenelse{%
    \(%
        \(\NOT\equal{#1}{\BooleanTrue}\)\OR%
        \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)\OR%
        \(\boolean{LWR@forcinghtmlpage}\)%
    \)%
    \AND%
    \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{FileDepth}}%
    \AND%
    \(%
        \NOT\boolean{CombineHigherDepths}\OR%
        \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{LWR@prevFileDepth}}%
    \)%
    \AND%


    \(% phantomsection
        \NOT\isempty{#3}%
        \OR%
        \(\NOT\equal{#1}{\BooleanTrue}\)%
    \)%
}%
```

If so: start a new HTML file:

```
{% new file
    \LWR@traceinfo{LWR@section: new HTML file}%
```

See if there was an optional TOC name entry:

```
    \IfNoValueTF{#2}%
```

If no optional entry

```
        {\LWR@newhtmlfile{#3}}%
```

If yes an optional entry

```
        {\LWR@newhtmlfile{#2}}%
}% new file
```

Else: No new HTML file:

```
{% not new file
```

Generate a new LaTeX page so that TOC and index page number points to the section:

```
   \LWR@traceinfo{LWR@section: not a new HTML file, about to LWR@orignewpage}%
    \LWR@maybe@orignewpage%
}% not new file

```

Remember this section's name for `\nameref`:

```
\IfValueT{#3}{%
```

```
    \LWR@traceinfo{LWR@section: about to LWR@setlatestname}%
    \IfValueTF{#2}{\LWR@setlatestname{#2}}{\LWR@setlatestname{#3}}%
}%
```

Print an opening comment with the level and the name; ex: “section” “Introduction” Footnotes may be used in section names, which would also appear in the HTML section opening comments, so the short TOC entry is used if possible, and a limited opening comment is made if the sectional unit is starred.

```
\begingroup%
\LWR@nullfonts%
\LWR@nullifyfootnotes%
\LWR@htmlcomment{%
    \LWR@orignewline%
    \IfValueTF{#2}%
        {...... #4 #2 ......}%
        {...... #4 #3 ......}%
    \LWR@orignewline%
}%
\LWR@orignewline%
\ifbool{HTMLDebugComments}%
    {%
        \IfBooleanTF{#1}% starred
            {%
                \IfNoValueTF{#2}% short TOC
                    {\LWR@htmlcomment{Opening #4* ‘‘#3’’}}%
                    {\LWR@htmlcomment{Opening #4* ‘‘#2’’}}%
            }%
            {%
                \IfNoValueTF{#2}% short TOC
                    {\LWR@htmlcomment{Opening #4 ‘‘#3’’}}%
                    {\LWR@htmlcomment{Opening #4 ‘‘#2’’}}%
            }%
        \LWR@orignewline%
    }%
    {}%
\endgroup%
```

For inline sections paragraph and subparagraph, start a new paragraph now:

```
\ifthenelse{%
    \cnttest{\@nameuse{LWR@depth#4}}{>=}{\LWR@depthparagraph}%
}%
    {\LWR@startpars}%
    {}%
```

Create the opening tag with an autosec:

```
\LWR@traceinfo{LWR@section: about to LWR@createautosec}%
\LWR@createautosec{\@nameuse{LWR@tag#4}}%

\setcounter{LWR@currentautosecpage}{\value{page}}%
```

Check if starred:

```
\IfBooleanTF{#1}%
{%
    \LWR@traceinfo{LWR@section: starred}%
```

Starred, but also forcing a TOC entry, so add unnumbered TOC name or regular name:

```
    \ifbool{LWR@forcinghtmltoc}%
    {%
        \addcontentsline{toc}{#4}{%
            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
        }%
    }%
    {}%
}% starred
```

Not starred, so step counter and add to TOC:

```
{% not starred
```

Only add a numbered TOC entry if section number is not too deep:

```
    \ifthenelse{%
        \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
    }%
    {% if secnumdepth
```

If in the main matter, step the counter and add the TOC entry. For `article` class, lwarp assumes that all is mainmatter.

```
        \LWR@traceinfo{LWR@section: about to test main matter}%
        \ifbool{LWR@mainmatter}%
        {%
            \LWR@traceinfo{LWR@section: yes mainmatter}%
            \refstepcounter{#4}%
```

Add main matter numbered TOC entry with the TOC name or the regular name:

```
            \LWR@traceinfo{LWR@section: about to addcontentsline}%
            \addcontentsline{toc}{#4}%
            {%
                \protect\numberline{%
                    \@nameuse{pre#4name}%
                    \@nameuse{the#4}%
                    \@nameuse{post#4name}%
                }%
                {%
                    \ignorespaces%
              \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}\protect\relax%
                }%
            }%
            \LWR@traceinfo{LWR@section: finished addcontentsline}%
        }% end of if main matter
```

If not main matter, add unnumbered TOC name or regular name:

```
        {% not main matter
            \LWR@traceinfo{LWR@section: no main matter}%
            \addcontentsline{toc}{#4}{%
                \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
            }%
        }% end of not main matter
    }% end of secnumdepth
```

Deeper than secnumdepth, so add an unnumbered TOC entry:

```
    {%
        \addcontentsline{toc}{#4}{%
            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
        }%
    }%
```

For part, print “Part”:

```
    \ifbool{LWR@mainmatter}%
    {%
        \ifthenelse{%
            \(\cnttest{\@nameuse{LWR@depth#4}}{<=}%
                {\value{secnumdepth}}\) \AND%
            \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)%
        }%
            {\@partnameformat}%
            {}%
```

Print the section number:

```
        \LWR@traceinfo{LWR@section: about to print section number}%
        \ifthenelse{%
            \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
        }%
            {%
                \ifstrequal{#4}{part}%
                {\protect\LWR@sectionnumber{\@partcntformat{#4}}}%
                {%
                    \ifstrequal{#4}{chapter}%
                        {%

                            \LWR@printchaptername%
                            \protect\LWR@sectionnumber{\@chapcntformat{#4}}%
                        }%
                        {\protect\LWR@sectionnumber{\@seccntformat{#4}}}%
                }%
            }%
            {}%
        \LWR@traceinfo{LWR@section: finished print section number}%
    }{}%
}% not starred
```

Print the section name:

```
\LWR@traceinfo{LWR@section: about to print the section name}%
\LWR@isolate{#3}%
```

Close the heading tag, such as /H2:

```
\LWR@traceinfo{LWR@section: about to close the heading tag}%
\LWR@htmltag{\@nameuse{LWR@tag#4end}}%
\LWR@orignewline%
```

Generate a LaTeX label.

Track the PDF page numbers of the HTML output. A new autopage label may be generated for LWR@currentautosecpage for the start of the section, and

also for the current page if it is different due to an SVG image in the section name. Also, the final page after the section has been created is updated in LWR@currentautosecfloatpage.

```
\LWR@traceinfo{LWR@section: about to create the LaTeX label}%
\setcounter{LWR@currentautosecfloatpage}{\value{page}}%
\LWR@newautopagelabel{LWR@currentautosecpage}\LWR@orignewline%
```

If this is the first section found in this file, create a label for prevous/next links:

```
\ifbool{LWR@setseqfilelabel}{}{%
    \label{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}%
    \booltrue{LWR@setseqfilelabel}%
}%
```

Start paragraph handing unless is an inline paragraph or subparagraph:

```
\ifthenelse{%
    \cnttest{\@nameuse{LWR@depth#4}}{<}{\LWR@depthparagraph}%
}%
    {\LWR@startpars}%
    {}%
```

If not starred, remember the previous depth to possibly trigger a new HTML page.

HOWEVER, allow a \part* to start a new HTML page. This is used by appendix.

A starred section does not trigger a new HTML page at the beginning of this macro, so it should not affect it here at the end either. This became an issue when a \listoftables was tested in the middle of the document. The \chapter* for the list was not allowing a new HTML page for the section following it while CombineHigherDepths was true.

```
\ifthenelse{%
    \NOT\equal{#1}{\BooleanTrue}\OR%
    \cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}%
}%
    {% not starred
        \setcounter{LWR@prevFileDepth}{\@nameuse{LWR@depth#4}}%
    }% not starred
    {}%
```

Reset to defaults if not a phantomsection:

```
\ifstrempty{#3}%
    {}%
    {%
        \global\boolfalse{LWR@forcinghtmlpage}%
        \global\boolfalse{LWR@forcinghtmltoc}%
    }%
%
\LWR@traceinfo{LWR@section: done}%
}
```

64.4 Pre- and post- sectioning names

\prebookname
\postbookname
Usually null, but is used by uj* and ut* Japanese classes.

```
\providecommand*{\prebookname}{}
\providecommand*{\postbookname}{}
```

\prepartname
\postpartname
Usually null, but is used by uj* and ut* Japanese classes.

```
\providecommand*{\prepartname}{}
\providecommand*{\postpartname}{}
```

\prechaptername
\postchaptername
Usually null, but is used by uj* and ut* Japanese classes.

```
\providecommand*{\prechaptername}{}
\providecommand*{\postchaptername}{}
```

\presectionname
\postsectionname
Always null, but provided here for algorithmic simplicity in \LWR@section.

```
\providecommand*{\presectionname}{}
\let\postsectionname\presectionname

\let\presubsectionname\presectionname
\let\postsubsectionname\postsectionname

\let\presubsubsectionname\presectionname
\let\postsubsubsectionname\postsectionname

\let\preparagraphname\presectionname
\let\postparagraphname\postsectionname

\let\presubparagraphname\presectionname
\let\postsubparagraphname\postsectionname
```

64.5 \section and friends

For memoir, a second optional argument is allowed.

For hypbmsec, a second optional argument or either parenthesis argument is allowed.

Each of these additional arguments are for headers or PDF bookmarks, and are ignored for HTML output.

\part * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\newcommand{\part@preamble}{}% for koma-script

\DeclareDocumentCommand{\part}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{part}%

    \part@preamble% for koma-script
    \renewcommand{\part@preamble}{}%
}
```

`\chapter` * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\let\@printcites\relax% for quotchap package

\newcommand{\chapter@preamble}{}% for koma-script

\@ifundefined{chapter}
{}
{%
    \DeclareDocumentCommand{\chapter}{s d() o o d() m}{%
        \LWR@section{#1}{#3}{#6}{chapter}%

        \@printcites% for quotchap package

        \chapter@preamble% for koma-script
        \renewcommand{\chapter@preamble}{}%
    }
}
```

`\section` * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\section}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{section}%
}
```

`\subsection` * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\subsection}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{subsection}%
}
```

`\subsubsection` * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\subsubsection}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{subsubsection}%
}
```

`\paragraph` * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\paragraph}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{paragraph}%
}
```

`\subparagraph` * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\subparagraph}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{subparagraph}%
}

\end{warpHTML}
```

65 Starting a new file

for HTML & PRINT:

```
\begin{warpall}
```

\HTMLLanguage Default language for the HTML lang tag.

```
\newcommand*{\LWR@currentHTMLLanguage}{en-US}

\newcommand*{\HTMLLanguage}[1]{%
    \renewcommand*{\LWR@currentHTMLLanguage}{#1}%
}
```

\theHTMLTitleSeparator May be used inside \theHTMLTitleSection to separate the website's overall HTML title and the particular page's section name.

```
\ifPDFTeX% pdflatex or dvi latex
    \ifdefstring{\inputencodingname}{utf8}{%
        \newcommand*{\theHTMLTitleSeparator}{ —\ }% EMdash
    }{%
        \newcommand*{\theHTMLTitleSeparator}{ -\ }% hyphen
    }%
\else%
    \ifpTeX
        \newcommand*{\theHTMLTitleSeparator}{ -\ }% hyphen
    \else
        \newcommand*{\theHTMLTitleSeparator}{ —\ }% EMdash
    \fi%
\fi%
```

\HTMLTitleBeforeSection Sets the HTML page's meta title tag to show the website title before the section name.

```
\newcommand*{\HTMLTitleBeforeSection}{%
    \def\theHTMLTitleSection{%
        \theHTMLTitle\theHTMLTitleSeparator\theHTMLSection%
    }%
}
```

\HTMLTitleAfterSection Sets the HTML page's meta title tag to show the section name before the website title.

```
\newcommand*{\HTMLTitleAfterSection}{%
    \def\theHTMLTitleSection{%
        \theHTMLSection\theHTMLTitleSeparator\theHTMLTitle%
    }%
}
```

\theHTMLTitleSection Forms the HTML page's meta title tag. The default is to show the website title before the section name.

```
\HTMLTitleBeforeSection
```

\theHTMLSection The section name is passed to \LWR@filestart, which then sets \theHTMLSection for use inside \theHTMLTitleSection to create an HTML meta title tag.

```
\newcommand*{\theHTMLSection}{}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@filestart [`⟨*section name*⟩`]` Creates the opening HTML tags.

```
\newcommand*{\LWR@filestart}[1][]{%
\LWR@traceinfo{LWR@filestart !#1!}%
```

Locally temporarily disable direct-formatting commands:

```
\begingroup%
\LWR@nullfonts%
```

Save the section name for use while creating the HTML meta `title` tag:

```
\edef\theHTMLSection{#1}%
```

Remove extra material:

```
\StrSubstitute{\theHTMLSection}{\protect}{\detokenize{-}}[\theHTMLSection]%
\StrSubstitute{\theHTMLSection}%
    {\detokenize{-----}}{\detokenize{-}}[\theHTMLSection]%
\StrSubstitute{\theHTMLSection}%
    {\detokenize{----}}{\detokenize{-}}[\theHTMLSection]%
\StrSubstitute{\theHTMLSection}%
    {\detokenize{---}}{\detokenize{-}}[\theHTMLSection]%
\StrSubstitute{\theHTMLSection}%
    {\detokenize{--}}{\detokenize{-}}[\theHTMLSection]%
```

If starts with a dash, remove the leading dash:

```
\IfBeginWith{\theHTMLSection}{\detokenize{-}}{%
    \StrGobbleLeft{\theHTMLSection}{1}[\theHTMLSection]%
}{}%
```

Create the page's HTML header:

```
\LWR@htmltag{!DOCTYPE html}\LWR@orignewline
```

The language is user-adjustable:

NOTE: `\LWR@orig@textquotedbl` is used here because `\textquotedbl` is nullified by `\LWR@nullfonts` while starting the new file.

```
\LWR@htmltag{%
  html lang=\LWR@orig@textquotedbl\LWR@currentHTMLLanguage\LWR@orig@textquotedbl%
}\LWR@orignewline
```

Start of the meta data:

```
\LWR@htmltag{head}\LWR@orignewline
```

Charset is fixed at UTF-8:

```
\LWR@htmltag{%
    meta charset=\LWR@orig@textquotedbl{}UTF-8\LWR@orig@textquotedbl\ /%
}\LWR@orignewline
```

Author:

```
\ifthenelse{\equal{\theHTMLAuthor}{}}%
    {}%
    {%
        \LWR@htmltag{%
        meta name=\LWR@orig@textquotedbl{}author\LWR@orig@textquotedbl\ % space
        content=\LWR@orig@textquotedbl\theHTMLAuthor\LWR@orig@textquotedbl\ /%
        }\LWR@orignewline%
    }%
```

lwarp is the generator:

```
\LWR@htmltag{%
    meta % space
    name=\LWR@orig@textquotedbl{}generator\LWR@orig@textquotedbl\ % space
   content=\LWR@orig@textquotedbl{}LaTeX Lwarp package\LWR@orig@textquotedbl\ /%
}\LWR@orignewline%
```

If there is a description, add it now:

```
\ifdefempty{\LWR@currentHTMLDescription}{}{%
    \LWR@htmltag{%
      meta name=\LWR@orig@textquotedbl{}description\LWR@orig@textquotedbl\ % space
      content=\LWR@orig@textquotedbl\LWR@currentHTMLDescription\LWR@orig@textquotedbl\ /%
    }\LWR@orignewline
}%
```

Mobile-friendly viewport:

```
\LWR@htmltag{%
    meta % space
    name=\LWR@orig@textquotedbl{}viewport\LWR@orig@textquotedbl\ % space
   content=\LWR@orig@textquotedbl{}width=device-width, initial-scale=1.0\LWR@orig@textquotedbl\ /%
}\LWR@orignewline
```

IE patch:

```
\LWR@htmltag{!-\/-[if lt IE 9]}\LWR@orignewline
\LWR@htmltag{%
    script % space
    src=\LWR@orig@textquotedbl{}%
        http://html5shiv.googlecode.com/svn/trunk/html5.js%
    \LWR@orig@textquotedbl%
}%
\LWR@htmltag{/script}\LWR@orignewline
\LWR@htmltag{![endif]-\/-}\LWR@orignewline
```

The page's title, if there is one. A section name is also added if given.

```
\ifthenelse{\equal{\theHTMLTitle}{}}%
    {}%
```

```
    {%
        \LWR@htmltag{title}%
        \ifdefempty{\theHTMLSection}%
            {\theHTMLTitle}%
            {\theHTMLTitleSection}%
        \LWR@htmltag{/title}\LWR@orignewline%
    }%
```

The page's stylesheet:

```
\LWR@htmltag{%
    link % space
    rel=\LWR@orig@textquotedbl{}stylesheet\LWR@orig@textquotedbl\ % space
    type=\LWR@orig@textquotedbl{}text/css\LWR@orig@textquotedbl\ % space
    href=\LWR@orig@textquotedbl\LWR@currentcss\LWR@orig@textquotedbl\ /%
}%
\LWR@orignewline
```

Optional MathJax support. The HTML tags must be turned off during the verbatim input, and the paragraph handling which was turned on at the end of verbatim input must be immediately turned off again.

```
\ifbool{mathjax}%
{%
    \begingroup%
    \LWR@restoreoriglists%
    \boolfalse{LWR@verbtags}%


        \IfFileExists{\LWR@mathjaxfilename}%
            {\verbatiminput{\LWR@mathjaxfilename}}%
            {%
                \PackageError{lwarp}%
                    {%
                \protect\MathJaxFilename\space specified the file\MessageBreak
                        \space\space\LWR@mathjaxfilename\MessageBreak
                        which does not exist%
                    }%
             {Specify an existing file, or remove \protect\MathJaxFilename.}%
            }%

    \booltrue{LWR@verbtags}%
    \endgroup%
    \LWR@stoppars%
}% end of mathjax
{}%
```

End of the header:

```
\LWR@htmltag{/head}\LWR@orignewline
```

Start of the body:

```
\LWR@htmltag{body}\LWR@orignewline


\endgroup%
\LWR@traceinfo{LWR@filestart: done}%
}
```

```
\end{warpHTML}
```

66 Starting HTML output

for HTML output:

```
\begin{warpHTML}
```

\LWR@LwarpStart Executed at the beginning of the entire document.

The use of \textquotedbl instead of " improves compatibility with xeCJK.

```
\catcode`\$=\active
\newcommand*{\LWR@LwarpStart}
{%
\LWR@traceinfo{LWR@lwarpStart}
```

If formatting for a word processor, force filedepth to single-file only, force HTML debug comments off.

```
\ifbool{FormatWP}{%
    \setcounter{FileDepth}{-5}%
    \boolfalse{HTMLDebugComments}%
}{}
```

Expand and detokenize \HomeHTMLFilename and \HTMLFilename:

```
\edef\LWR@strresult{\HomeHTMLFilename}
\edef\HomeHTMLFilename{\detokenize\expandafter{\LWR@strresult}}
\edef\LWR@strresult{\HTMLFilename}
\edef\HTMLFilename{\detokenize\expandafter{\LWR@strresult}}
```

Force onecolumn and empty page style:

```
\LWR@origonecolumn%
\LWR@origpagestyle{empty}%
```

No black box for overfull lines:

```
\overfullrule=0pt
```

Reduce chance of line overflow when HTML tags are added:

```
\LWR@print@footnotesize%
```

In PDF output, don't allow line breaks to interfere with HTML tags:

```
\LWR@print@raggedright%
\LetLtxMacro{\\}{\LWR@endofline}%
```

Spread the lines for *pdftotext* to read them well:

```
\linespread{1.3}%
```

For *pdftotext* to reliably identify paragraph splits:

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{2ex}
```

For the lateximage record file:

```
\immediate\openout\LWR@lateximagesfile=\BaseJobname-images.txt
```

Removes space around the caption in the HTML:

```
\setlength{\belowcaptionskip}{0ex}
\setlength{\abovecaptionskip}{0ex}
```

Redefine the plain page style to be empty when used by index pages:

```
\renewcommand{\ps@plain}{}
```

Plug in some new actions. This is done just before the document start so that they won't be over-written by some other package.

Float captions:

```
\let\LWR@origcaption\caption
```

Not yet started any paragraph handling:

```
\global\booltrue{LWR@doingparhooks}
\global\boolfalse{LWR@doingapar}
\global\boolfalse{LWR@doingstartpars}
```

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

```
\let\color@endgroup\endgroup
```

Document and page settings:

```
\mainmatter
\LWR@origpagenumbering{arabic}
```

Start a new HTML file and a header:

```
\LWR@traceinfo{LWR@lwarpStart: Starting new file.}
\LWR@filestart%
```

Tell *lwarpmk* that the lwarp package is in use. This allows *lwarpmk* to warn if usepackage{lwarp} was somehow disabled.

```
\begingroup%
\LWR@nullfonts%
\LWR@htmlblockcomment{%
|Using lwarp|%
\LWR@htmlsectionfilename{\LWR@thisfilename}|%
}
\endgroup%


\LWR@traceinfo{LWR@lwarpStart: Generating first header.}
\ifdefempty{\LWR@firstpagetop}{}{%
    \LWR@htmltag{header}\LWR@orignewline
    \LWR@startpars
    \LWR@firstpagetop
    \LWR@stoppars
```

```
    \LWR@htmltag{/header}\LWR@orignewline
}%

\LWR@htmlelementclass{div}{bodywithoutsidetoc}
\LWR@htmlelementclass{main}{bodycontainer}
\LWR@traceinfo{LWR@lwarpStart: Generating textbody.}
\LWR@htmlelementclass{section}{textbody}
```

Create a label for previous/next links, and remember it has been done:

```
\booltrue{LWR@setseqfilelabel}%
\label{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}
```

Patch the `itemize`, `enumerate`, and `description` environments and `\item`. This works with the native LaTeX environments, as well as those provided by enumitem, enumerate, and paralist.

```
\LWR@patchlists
```

Ensure that math mode is active to call lwarp's patches:

```
\catcode`\$=\active
```

Required for `\nameref` to work with SVG math:

```
\immediate\write\@mainaux{\catcode`\string$\active}%
\LetLtxMacro\LWR@syntaxhighlightone$% balance for editor syntax highlighting
```

Allow HTML paragraphs to begin:

```
\LWR@startpars
```

If using MathJax, disable `\ensuremath` by printing a nullified definition at the start of each file, and add further customizations:

```
\ifbool{mathjax}{
    \typeout{---}
    \typeout{Package lwarp:}
    \typeout{Processing MathJax customizations for the first HTML page.}
    \typeout{Later HTML pages will take the same amount of time.}
   \typeout{If this takes too long, see the Lwarp manual regarding customizing MathJax.}
}{}

\LWR@customizeMathJax

\ifbool{mathjax}{
    \typeout{Done.}
    \typeout{---}
}{}
```

First autopage label in case a figure occurs early before the first section: A new autopage label may be generated for `LWR@currentautosecpage` for the start of the section, and also for the current page if it is different due to an SVG image in the section name. Also, the final page after the section has been created is updated in `LWR@currentautosecfloatpage`.

```
\setcounter{LWR@currentautosecfloatpage}{\value{page}}%
\LWR@newautopagelabel{LWR@currentautosecpage}%
```

```
\LWR@traceinfo{LWR@lwarpStart: done}
}
\catcode`\$=3% math shift until lwarp starts

\end{warpHTML}
```

67 Ending HTML output

for HTML output:
```
\begin{warpHTML}
```

`\LWR@requesttoc` {⟨*boolean*⟩} {⟨*suffix*⟩} Requests that a TOC, LOF, or LOTbe generated.

```
\newcommand*{\LWR@requesttoc}[2]{%
\ifbool{#1}
{
    \expandafter\newwrite\@nameuse{tf@#2}
    \immediate\openout \@nameuse{tf@#2} \jobname.#2\relax
}{}
}
```

`\LWR@LwarpEnd` Final stop of all HTML output:

```
\newcommand*{\LWR@LwarpEnd}
{
\LWR@stoppars
\LWR@closeprevious{finished}
```

At the bottom of the ending file:

Print any pending footnotes:

```
\LWR@printpendingfootnotes
```

Close the textbody.

(The `\LWR@origtilde` is in case no autopage is required for the label, which would not print anything, and something must be printed before the newline.)

```
\label{\BaseJobname-autofile-last}\LWR@origtilde\LWR@orignewline


\LWR@htmlelementclassend{section}{textbody}
\LWR@htmlelementclassend{main}{bodycontainer}
\LWR@htmlelementclassend{div}{bodyandsidetoc}
```

Create the footer if not EPUB

```
\ifbool{FormatEPUB}{}{\LWR@createfooter}
```

No bottom navigation if are finishing the home page, or if formatting for an EPUB or word processor.

Presumably has a table-of-contents.

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {
        \ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}
    }

\LWR@stoppars% final stop of all paragraphs
```

Finish the HTML file:

```
\LWR@htmltag{/body}\LWR@orignewline
\LWR@htmltag{/html}\LWR@orignewline
```

Seems to be required sometimes:

```
\LWR@maybe@orignewpage
}
```

enddocument/info (*Hook*) [LaTeX] Used to close the `*-images.txt` file.

\enddocument If labels have not changed, mark successful completion of the `lateximages.txt` file. Executed as everything is being shut down.

For the newer kernel hooks, see **`texdoc lthooks-doc`** and **`texdoc ltshipout-doc`**.

```
\ifdef{\AddToHook}{% newer kernel
    \AddToHook{enddocument/info}{%
       \if@filesw
       \ifx \@multiplelabels \relax
         \if@tempswa
```

This is where warnings of duplicate labels would appear.

```
         \else
```

No duplicate labels, so safe to create images.

```
            \immediate\write\LWR@lateximagesfile{%
                |end|end|end|%
            }%
         \fi
       \fi\fi
    }
}% newer kernel
{% older kernel
    \xpatchcmd{\enddocument}
        {%
            \if@tempswa
            \@latex@warning@no@line{Label(s) may have changed.
            Rerun to get cross-references right}%
            \fi
        }
        {%
            \if@tempswa
                \@latex@warning@no@line{Label(s) may have changed.
                Rerun to get cross-references right}%
            \else
```

No duplicate labels, so safe to create images.

```
                \immediate\write\LWR@lateximagesfile{%
                    |end|end|end|%
                }%
            \fi
        }
        {}
        {
            \AtEndDocument{
                \PackageWarningNoLine{lwarp}
                {%
                    Could not patch \protect\enddocument.\MessageBreak
                  If labels have changed, be sure to recompile before\MessageBreak
                    creating lateximages with\MessageBreak
                    \space\space lwarpmk limages,\MessageBreak
                    or the images may be corrupt%
                }
            }
        }
}% older kernel
```

68 Nullifying foreground/background hooks

See **texdoc lthooks-doc** and **textdoc ltshipout-doc**.

shipoout/background (*Hook*) [LaTeX]

Nullified.

shipoout/foreground (*Hook*) [LaTeX]

Nullified.

```
\ifdef{\RemoveFromHook}{
    \AfterEndPreamble{
        \IfHookEmptyTF{shipout/background}{}{
            \PackageInfo{lwarp}{Removing background hook}
            \RemoveFromHook{shipout/background}[*]
        }
        \IfHookEmptyTF{shipout/foreground}{}{
            \PackageInfo{lwarp}{Removing foreground hook}
            \RemoveFromHook{shipout/foreground}[*]
        }
    }
}{}
```

```
\end{warpHTML}
```

69 Title page

package support
⚠ load order

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 69.8.

affiliation

lwarp provides for the \author macro an additional \affiliation macro to pro-

vide an affiliation and other additional information for each author in the title page. The affiliation information is removed when using titlingpage's `\theauthor` in the main text.

reusing titlepage information

The titling package maintains the definitions of `\thetitle`, `\theauthor`, etc., after the title has been typeset. These commands are to be used to refer to the document's title and author, etc., in the main text. These definitions have the `\thanks` and `\affiliation` removed, and for `\author` the `\and` is replaced to generate a simple inline list of authors separated by commas. Note: `\theauthor` does not work well with authblk unless the traditional LaTeX syntax is used.

⚠ `\theauthor`, authblk

custom titlepages

`\printtitle`, `\printauthor`, etc., are provided for use inside a custom `titlepage` or `titlingpage` environment, and these retain the `\thanks` and `\affiliation`.

`\printthanks`

`\printthanks` has been added to force the printing of thanks inside a `titlingpage` environment when `\maketitle` is not used.

⚠ `\thanks`

Inside a `\titlepage` or `\titlingpage` environment, use `\thanks` instead of `\footnote` for acknowledgements, etc.

69.1 Setting the title, etc.

The following provide setting commands for both HTML and print outputs.

`\author` `\and`

`{`⟨*author*⟩`}` While using `\maketitle` and print mode, the author is treated as a single-column `tabular` and the `\and` feature finishes the current `tabular` then starts a new one for the next author. Each author thus is placed into its own `tabular`, and an affiliation may be placed on its own line such as

```
\author{Name \\ Affiliation \and Second Name \\ Second Affiliation}
```

For HTML, the entire author block is placed inside a `<div>` of class `author`, and each individual author is inside a `<div>` of class `oneauthor`.

`\@title` `\@author` `\@date`

`\@title`, `\@author`, and `\@date` store the values as originally assigned, including any `\thanks`, `\and`, or `\affiliation`. These are low-level macros intended to be used by other macros only inside a `titlepage` or `titlingpage`, and are used by `\maketitle`. The author is printed inside a single-column `tabular`, which becomes multiple single-column `tabulars` if multiples authors are included. For HTML, these `tabulars` become side-by-side `<div>`s of class `oneauthor`, all of which are combined into one `<div>` of class `author`.

`\printtitle` `\printauthor` `\printdate`

`\printtitle`, etc. are user-level macros intended to be used in custom `titlepage` or `titlingpage` environments in cases where `\maketitle` is not desired. These commands preserve the `\thanks`, etc., and should not be used in the main text.

`\thetitle` `\theauthor` `\thedate` `\HTMLPageBottom`

`\thetitle`, `\theauthor`, and `\thedate` are available if titling has been loaded, and are sanitized user-level versions from which have been removed the `\thanks` and `\affiliation`, and `\and` is changed for inline text usage. The author is printed inline without `\affiliation` or `\thanks`, with `\and` placing commas between multiple authors. Thus, these commands are to be used in the main text whenever the user wishes to refer to the document's title and such. One practical use for this is to place the authors at the bottom of each HTML page, such as:

```
\HTMLPageBottom{
```

```
  \begin{center}\textcopyright~20xx \theauthor\end{center}
}
```

⚠ \theauthor, authblk

\theauthor does not work well if authblk is used. If \theauthor is important, it is recommended to use the standard LaTeX syntax for \author, optionally with lwarp's \affiliation macro as well.

⚠ affiliations

After \maketitle has completed, \theauthor retains the definition of the author, but \and is changed to become a comma and a space, intending to print the authors names separated by spaces. This fails when affiliations are included on their own table rows.

\affiliation

A solution, provide here, is to define a macro \affiliation which, during \maketitle, starts a new row and adds the affiliation, but after \maketitle is finished \affiliation is re-defined to discard its argument, thus printing only the author names when \author is later used inline.

69.2 \if@titlepage

for HTML & PRINT:

```
\begin{warpall}
```

\if@titlepage

Some classes do not provide \if@titlepage. In this case, provide it and force it false.

```
\ifcsvoid{@titlepagefalse}{
    \newif\if@titlepage
    \@titlepagefalse
}{}

\end{warpall}
```

69.3 Changes for \affiliation

\affiliation {⟨*text*⟩}

Adds the affiliation to the author for use in \maketitle.

Inside titlepage, this macro prints its argument. Outside, it is null.

for HTML & PRINT:

```
\begin{warpall}
\providerobustcmd{\affiliation}[1]{}
\end{warpall}
```

for PRINT output:

```
\begin{warpprint}

\AtBeginEnvironment{titlepage}{
\renewrobustcmd{\affiliation}[1]{\\ \textsc{\small#1}}
}

\AtBeginDocument{
\IfPackageLoadedTF{titling}{
\AtBeginEnvironment{titlingpage}{
\renewrobustcmd{\affiliation}[1]{\\ \textsc{\small#1}}
```

```
}
}{}% titling loaded
}% AtBeginDocument

\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

`titlepage` (*env.*) Sets up a of class `titlepage`. Provided even for memoir class, since it is used by `\maketitle`.

```
\DeclareDocumentEnvironment{titlepage}{}
{%
    \renewrobustcmd{\affiliation}[1]{\\ \InlineClass{affiliation}{##1}}%
    \LWR@printpendingfootnotes
    \LWR@forcenewpage
    \BlockClass{titlepage}
}
{
    \endBlockClass
    \LWR@printpendingfootnotes
}

\end{warpHTML}
```

69.4 Printing the thanks

`\printthanks` Forces the `\thanks` to be printed. This is necessary in a `titlingpage` environment when `\maketitle` was not used.

for PRINT output:

```
\begin{warpprint}
\newcommand*{\printthanks}{\@thanks}
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
\newcommand*{\printthanks}{\LWR@stoppars\@thanks\LWR@startpars}
\end{warpHTML}
```

69.5 Printing the title, etc. in HTML

The following are for printing the title, etc. in a `titlepage` or a `titlingpage` in HTML:

for HTML output:

```
\begin{warpHTML}
```

`\printtitle`

```
\newcommand*{\printtitle}
{%
    \LWR@stoppars%
    \LWR@htmltag{\LWR@tagtitle}%
    \@title%
    \LWR@htmltag{\LWR@tagtitleend}%
    \LWR@startpars%
}
```

\LWR@printthetitle A private version which prints the title without footnotes, used to title each HTML page.

```
\newcommand*{\LWR@printthetitle}
{%
    \LWR@stoppars%
    \LWR@htmltag{\LWR@tagtitle}%
    \thetitle%
    \LWR@htmltag{\LWR@tagtitleend}%
    \LWR@startpars%
}
```

\printauthor HTML version.

```
\newcommand*{\printauthor}{
```

The entire author block is contained in a named author:

```
\begin{BlockClass}{author}
```

\and finishes one author and starts the next:

```
\renewcommand{\and}{%
\end{BlockClass}
\begin{BlockClass}{oneauthor}
}
```

Individual authors are contained in a named oneauthor:

```
\begin{BlockClass}{oneauthor}
\@author
\end{BlockClass}
\end{BlockClass}
}
```

\printdate

```
\newcommand*{\printdate}{%
\begin{BlockClass}{titledate}
\@date
\end{BlockClass}
}
```

```
\end{warpHTML}
```

69.6 Printing the title, etc. in print form

The following are for printing the title, etc. in a titlepage or a titlingpage in print form:

for PRINT output:

```
\begin{warpprint}
```

\printtitle

```
\newcommand*{\printtitle}{{\Huge\@title}}
```

\printauthor Print mode.

```
\newcommand*{\printauthor}
    {{\large\begin{tabular}[t]{c}\@author\end{tabular}}}
```

\printdate

```
\newcommand*{\printdate}{{\small\textit{\@date}}}
```

```
\end{warpprint}
```

69.7 \maketitle for HTML output

An HTML of class titlepage is used.

\thanks are a form of footnotes used in the title page. See section 60 for other kinds of footnotes.

See \thanksmarkseries{series}, below, to set the style of the footnote marks.

for HTML output:

```
\begin{warpHTML}
```

```
\IfClassLoadedTF{memoir}
{
\newcommand{\LWR@setfootnoteseries}{%
    \renewcommand\thefootnote{\@arabic\c@footnote}%
}
}{% not memoir
\if@titlepage
\newcommand{\LWR@setfootnoteseries}{%
    \renewcommand\thefootnote{\@arabic\c@footnote}%
}
\else
\newcommand{\LWR@setfootnoteseries}{%
    \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
}
\fi
}% not memoir
```

\LWR@maketitlesetup Patches \thanks macros.

```
\newcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
\LWR@setfootnoteseries%
\def\@makefnmark{%
    \textsuperscript{\thefootnote}%
}
```

\thefootnote ⇒ \nameuse{arabic}{footnote}, or
\thefootnote ⇒ \nameuse{fnsymbol}{footnote}

Redefine the footnote text:

```
\long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
\textsuperscript{\@thefnmark}~%
```

\makethanksmark ⇒ \thanksfootmark ⇒ \tamark ⇒ \@thefnmark ⇒ \itshape a (or similar)

Print the text:

```
{##1}%
}%
}
```

`\@fnsymbol {⟨counter⟩}`

Re-defined to use an HTML entity for the double vertical bar symbol. The original definition used `\|` which was not being seen by *pdftotext*.

```
\def\LWR@HTML@@fnsymbol#1{%
    \ifcase#1\or *\or
    \HTMLentity{dagger}\or
    \HTMLentity{Dagger}\or
    \HTMLentity{sect}\or
    \HTMLentity{para}\or
    \HTMLunicode{2016}\or
    **\or
    \HTMLentity{dagger}\HTMLentity{dagger} \or
    \HTMLentity{Dagger}\HTMLentity{Dagger} \else
    \@ctrerr\fi%
}
\LWR@formatted{@fnsymbol}
```

`\maketitle` HTML mode. Creates an HTML titlepage div and typesets the title, etc.

Code from the titling package is adapted, simplified, and modified for HTML output.

The name `\LWR@maketitle` is used to preserve its definition in case a later package overwrites `\maketitle`.

```
\newcommand*{\LWR@maketitle}{%
```

An HTML titlepage is used for all classes.

```
\begin{titlepage}
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any `\thanks` footnotes:

```
\LWR@stoppars\@thanks\LWR@startpars
```

Close the HTML titlepage div and cleanup:

```
\end{titlepage}
\setcounter{footnote}{0}%
\global\let\thanks\relax
\global\let\maketitle\relax
\global\let\@maketitle\relax
\global\let\@thanks\@empty
\global\let\@author\@empty
\global\let\@date\@empty
\global\let\@title\@empty
\global\let\title\relax
\global\let\author\relax
\global\let\date\relax
\global\let\and\relax
}

\LetLtxMacro\maketitle\LWR@maketitle
```

`\@maketitle` HTML mode. Typesets the title, etc.:

```
\providecommand*{\@maketitle}{}
\renewrobustcmd{\@maketitle}{%
    \LWR@stoppars%
    \LWR@htmltag{\LWR@tagtitle}%
    \@title%
    \LWR@htmltag{\LWR@tagtitleend}%
    \LWR@startpars%
    \begin{BlockClass}{author}%
```

For IEEEtran class:

```
    \renewcommand*{\cr}{}%
    \renewcommand*{\crcr}{}%
    \renewcommand*{\noalign}{}%

        \renewcommand{\and}{%
            \end{BlockClass}%
            \begin{BlockClass}{oneauthor}%
        }%
        \begin{BlockClass}{oneauthor}%
            \@author%
        \end{BlockClass}%
    \end{BlockClass}%
    \begin{BlockClass}{titledate}%
    \@date%
    \end{BlockClass}%
}
```

`\LWR@titlingmaketitle` `\maketitle` for use inside an HTML `titlingpage` environment.

```
\newcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
\LWR@stoppars\@thanks\LWR@startpars
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any \thanks footnotes:

```
\LWR@stoppars\@thanks\LWR@startpars
}
```

```
\end{warpHTML}
```

69.8 \published and \subtitle

\subtitle and \published

To add \subtitle and \published to the titlepage, load the titling package and use \AddSubtitlePublished in the preamble.

The default lwarp.css has definitions for the published and subtitle classes.

If titling is loaded, \AddSubtitlePublished creates a number of additional macros, and also assigns some of the titling hooks. If titling is not loaded, \AddSubtitlePublished creates null macros.

⚠ titling hooks

Do not use \AddSubtitlePublished if the user has patched the titling hooks for some other reason. Portions are marked \warpprintonly to reduce extra tags in HTML. Similarly, BlockClass has no effect in print mode. Thus, the following may be marked warpall.

for HTML & PRINT:

```
\begin{warpall}
```

\AddSubtitlePublished

Adds \published and \subtitle, and related.

```
\newcommand*{\AddSubtitlePublished}{%
\IfPackageLoadedTF{titling}{% yes titling package
    \newcommand{\@published}{}%
    \newcommand{\published}[1]{\gdef\@published{##1}}%
    \renewcommand*{\maketitlehooka}{\printpublished}%
    \newcommand*{\printpublished}{%
        \warpprintonly{\begin{center}\unskip}%
        \begin{BlockClass}{published}%
        \warpprintonly{\large\itshape}%
        \@published%
        \end{BlockClass}%
        \warpprintonly{\end{center}}%
    }%
    \newcommand{\@subtitle}{}%
    \newcommand{\subtitle}[1]{\gdef\@subtitle{##1}}%
    \renewcommand*{\maketitlehookb}{\printsubtitle}%
    \newcommand*{\printsubtitle}{%
        \warpprintonly{\begin{center}\unskip}%
        \begin{BlockClass}{subtitle}%
        \warpprintonly{\Large\itshape}%
        \@subtitle%
```

```
        \end{BlockClass}%
        \warpprintonly{\end{center}}%
    }%
}% yes titling package
{% no titling package

    \def\@published{}%
    \DeclareDocumentCommand{\published}{m}{\gdef\@published{##1}}%
    \DeclareDocumentCommand{\printpublished}{}{}%
    \def\@subtitle{}%
    \DeclareDocumentCommand{\subtitle}{m}{\gdef\@subtitle{##1}}%
    \DeclareDocumentCommand{\printsubtitle}{}{}%
}% no titling package
}% \AddSubtitlePublished

\end{warpall}
```

70 Abstract

The following code replaces the LaTeX default, and will itself be replaced later if the abstract package is loaded.

for HTML output:

```
\begin{warpHTML}
```

`\abstractname` User-redefinable title for the abstract.

Also over-written by the babel package.

```
\providecommand*{\abstractname}{Abstract}
```

Some classes allow an optional name, so it is allowed here.

`abstract` (*env.*)

```
\DeclareDocumentEnvironment{abstract}{O{\abstractname}}
{
    \LWR@forcenewpage
    \BlockClass{abstract}
    \BlockClassSingle{abstracttitle}{#1}
}
{
    \endBlockClass
}

\end{warpHTML}
```

71 Quote and verse

71.1 Attributions

`\attribution` {⟨*name*⟩}

For use with quote, quotation, verse:

Ex: `"A quotation." \attribution{\textsc{Author Name}\\\textsl{Book Title}}`

for HTML & PRINT:

```
\begin{warpall}
\newcommand{\attribution}[1]{
    \begin{flushright}
    \unskip
    #1
    \end{flushright}%
}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
\newcommand{\LWR@HTML@attribution}[1]{%
    \LWR@stoppars%
    \begin{BlockClass}{attribution}
    #1
    \end{BlockClass}
    \LWR@startpars%
}
\LWR@formatted{attribution}
\end{warpHTML}
```

71.2 Quotes, quotations

for HTML output:

```
\begin{warpHTML}
```

quote (*env.*)

```
\newenvironment*{LWR@HTML@quote}
{
    \LWR@forcenewpage
    \LWR@htmlblocktag{blockquote}
}
{\LWR@htmlblocktag{/blockquote}}

\LWR@formattedenv{quote}
```

quotation (*env.*)

```
\newenvironment*{LWR@HTML@quotation}
{
    \LWR@forcenewpage
    \LWR@htmlblocktag{blockquote}
}
{\LWR@htmlblocktag{/blockquote}}

\LWR@formattedenv{quotation}
```

```
\end{warpHTML}
```

71.3 Verse

When using verse or memoir, always place a \\ after each line.

\attrib

The documentation for the verse and memoir packages suggest defining an \attrib command, which may already exist in current documents, but it will only work for print output. lwarp provides \attribution, which works for both print and HTML output. To combine the two so that \attrib is used for print and \attribution is used for HTML:

```
\begin{warpHTML}
\let\attrib\attribution
\end{warpHTML}
```

\vleftskip (*Len*)
\vleftmargini (*Len*)
\HTMLvleftskip (*Len*)
\HTMLleftmargini (*Len*)

These lengths are used by verse and memoir to control the left margin, and they may already be set by the user for print output. New lengths \HTMLvleftskip and \HTMLleftmargini are provided to control the margins in HTML output. These new lengths may be set by the user before any verse environment, and persist until they are manually changed again. One reason to change \HTMLleftmargini is if there is a wide \flagverse in use, such as the word "Chorus", in which case the value of \HTMLleftmargini should be set to a wide enough length to contain "Chorus". The default is wide enough for a stanza number.

⚠ spacing

Horizontal spacing relies on *pdftotext*'s ability to discern the layout (-layout option) of the text in the HTML-tagged PDF output. For some settings of \HTMLleftmargini or \HTMLleftskip the horizontal alignment may not work out exactly, in which case a label may be shifted by one space. During translation to HTML, the stanza numbers are kept out of the left margin, which would have caused *pdftotext* to shift everything over.

⚠ verse margin

71.3.1 LaTeX core verse environment

for HTML output:

```
\begin{warpHTML}
```

verse (*env.*)

```
\newenvironment{LWR@HTML@verse}
               {\let\\\newline% lwarp
                \list{}{\itemsep      \z@
                        \itemindent   -1.5em%
                        \listparindent\itemindent
                        \rightmargin  \leftmargin
                        \advance\leftmargin 1.5em}%
                \item\relax}
               {\endlist}

\LWR@formattedenv{verse}
```

```
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

71.3.2 verse and memoir

The following lengths are used by verse and memoir. They may be set in either print or HTML output, but are only used in HTML. This allows the user to set \vleftskip and \leftmargini for print output, and optionally select different values for HTML.

\HTMLvleftskip (*Len*) Sets \vleftskip inside a verse environment in HTML.

```
\newlength{\HTMLvleftskip}
\setlength{\HTMLvleftskip}{1em}
```

\HTMLleftmargini (*Len*) Sets \leftmargini inside a verse environment in HTML.

```
\newlength{\HTMLleftmargini}
\setlength{\HTMLleftmargini}{4.5em}

\end{warpall}
```

72 Verbatim and tabbing

for HTML & PRINT:
```
\begin{warpall}
```

\VerbatimHTMLWidth (*Len*) Width to use in HTML Verbatim environment.

This width is used when placing line numbers to the right. Ignored during print output.

```
\newlength{\VerbatimHTMLWidth}
\setlength{\VerbatimHTMLWidth}{4in}
\end{warpall}
```

for HTML output:
```
\begin{warpHTML}
```

LWR@verbtags (*bool*) Used to temporarily turn off verbatim tags while doing \verbatiminput in the HTML head.

```
\newbool{LWR@verbtags}
\booltrue{LWR@verbtags}
```

\verb Patched to encapsulate the verbatim text inside span with a class of texttt.

```
\LetLtxMacro\LWR@orig@verb@egroup\verb@egroup

\def\LWR@verb@egroup@endspan{%
    \LWR@orig@verb@egroup%
    \LWR@htmltag{/span}%
    \endgroup%
}


\xpretocmd{\verb}
    {%
        \begingroup%
        \LWR@htmltag{span class=\textquotedbl{}texttt\textquotedbl}%
        \let\verb@egroup\LWR@verb@egroup@endspan%
    }
    {}
    {\LWR@patcherror{LaTeX}{verb}}
```

\LWR@atbeginverbatim [⟨*1: style*⟩] {⟨*2: class*⟩}

Encloses a verbatim environment with the given CSS class.

The use of `\textquotedbl` instead of " improves compatibility with xeCJK.

```
\newcommand*{\LWR@atbeginverbatim}[2][]
{%
```

Stop generating HTML paragraph tags:

```
\LWR@stoppars%
```

Avoid excessive space between lines:

```
\setlength{\parskip}{0ex}%
\setlength{\topsep}{0pt}%
\setlength{\partopsep}{0pt}%
```

Inside the verbatim, temporarily prevent underfull `\hbox` warnings.

```
            \hbadness=10000\relax%
```

Create a new `pre` of the given class. The tags may temporarily be turned off for internal use, such as loading the MathJax script.

```
\ifbool{LWR@verbtags}{%
    \LWR@htmltag{pre class=\textquotedbl#2\textquotedbl%
        \ifthenelse{\equal{#1}{}}{}{ style=\textquotedbl#1\textquotedbl}%
    }%
    \par%
}{}%
```

Use a mono-spaced font to preserve horizontal positioning. If horizontal alignment is important for the user, use a mono-spaced font in the CSS for the `verse` class.

```
\begingroup%

\LWR@print@normalfont%
\LWR@origttfamily%
\LWR@print@scriptsize%
```

Since inside a <pre>, restore the original list processing:

```
\LWR@restoreoriglists%
```

Turn off babel-french extra space before punctuation:

```
\LWR@hook@processingtags%
```

Do not produce HTML tags for `\hspace` inside a verse `par`. Restore plain LaTeX `\hspace` functionality:

```
\let\hspace\LWR@print@hspace%
}
```

`\LWR@afterendverbatim` Finishes enclosing a verbatim environment.

```
\newcommand*{\LWR@afterendverbatim}{%
\endgroup%
\par%
```

At the end of the environment, close the pre:

```
\ifbool{LWR@verbtags}{%
    \noindent\LWR@htmltag{/pre}\par% pre
}{}%
```

Resume regular paragraph handling:

```
\LWR@startpars%
}
```

`\verbatiminput {`⟨*filename*⟩`}`

Patch `\verbatiminput` to add HTML tags:

```
\newcommand{\LWR@HTML@verbatim@input}[2]{%
    \ifbool{LWR@verbtags}{\LWR@forcenewpage}{}%
    \LWR@atbeginverbatim{Verbatim}%
    \LWR@print@verbatim@input{#1}{#2}%
    \LWR@afterendverbatim%
}

\LWR@formatted{verbatim@input}
```

`verbatim` (*env.*)

```
\AfterEndPreamble{
\LWR@traceinfo{Patching verbatim.}
\AtBeginEnvironment{verbatim}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
            \LWR@forcenewpage%
            \LWR@atbeginverbatim{verbatim}%
        }%
}
\AfterEndEnvironment{verbatim}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
            \LWR@afterendverbatim%
        }%
}
%
\AtBeginEnvironment{verbatim*}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
            \LWR@forcenewpage%
            \LWR@atbeginverbatim{verbatim}%
        }%
}
\AfterEndEnvironment{verbatim*}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
```

```
        {}%
        {%
            \LWR@afterendverbatim%
        }%
}
}
```

tabbing (*env.*) The tabbing environment works, except that SVG math and lateximages do not yet work inside the environment.

⚠ math in tabbing If math is used inside tabbing, place tabbing inside a lateximage environment, which will render the entire environment as a single SVG image.

```
\newenvironment*{LWR@HTML@tabbing}
{%
    \LWR@forcenewpage%
    \LWR@atbeginverbatim{tabbing}%
    \let\enskip\LWR@print@enskip%
    \let\quad\LWR@print@quad%
    \let\qquad\LWR@print@qquad%
    \let~\LWR@origtilde%
    \let\,\LWR@origcomma%
    \let\thinspace\LWR@print@thinspace%
    \let\negthinspace\LWR@print@negthinspace%
    \LWR@print@tabbing%
}
{%
    \endLWR@print@tabbing%
    \LWR@afterendverbatim%
}

\LWR@formattedenv{tabbing}
```

```
\end{warpHTML}
```

73 Theorems

\newtheorem {⟨*text*⟩} [⟨*counter*⟩] — *or* — [⟨*oldname*⟩] {⟨*text*⟩}

A few minor changes are made to supply HTML tags.

- The entire theorem is placed into a of class theoremcontents.
- The label for each theorem is placed inside a of class theoremlabel.
- The contents are placed inside a of class theoremcontents.

for HTML output:

```
\begin{warpHTML}
```

\@begintheorem {⟨*name*⟩} {⟨*number*⟩}

```
\renewcommand{\@begintheorem}[2]{%
\LWR@forcenewpage
```

```
8498     \LWR@printpendingfootnotes%                    lwarp

8499 \BlockClass{theoremcontents}
8500 \trivlist
8501 \item[\InlineClass{theoremlabel}{#1\ #2\ }]\itshape
8502 }
```

\@opargbegintheorem {⟨*name*⟩} {⟨*number*⟩} {⟨*oparg*⟩}

LaTeX defines this, but amsthm \relaxes it, so it will not be defined if amsthm is loaded before lwarp.

```
8503 \ifundef{\@opargbegintheorem}{}{
8504     \renewcommand{\@opargbegintheorem}[3]{%
8505         \LWR@forcenewpage
8506         \BlockClass{theoremcontents}
8507         \trivlist
8508         \item[\InlineClass{theoremlabel}{#1\ #2\ (#3)\ }]\itshape
8509     }
8510 }
```

\@endtheorem

```
8511 \renewcommand*{\@endtheorem}{%
8512 \endtrivlist

8513     \LWR@printpendingfootnotes%                    lwarp

8514 \endBlockClass% theoremcontents
8515 }

8516 \end{warpHTML}
```

74 Lists

The environments itemize, enumerate, and description are patched when lwarp is started. These patches support the standard LaTeX environments, as well as those of enumerate, enumitem, and paralist, and at least the French version of babel. Additional patches are done on a package-specific basis.

The LaTeX source for itemize and enumerate are found in source2e, but the source for description is found in article.cls, etc.

empty item

To have an empty item,use \mbox{} or a trailing backslash. This forces a new line in print output, matching the new line which will appear in HTML output. Ex:

```
begin{itemize}
item \mbox{}
    \begin{itemize}
...
    \end{itemize}
item \
```

```
  \begin{itemize}
...
  \end{itemize}
```

\makelabel While inside a list environment, lwarp nullifies a number of TEX horizontal skip and fill commands, allowing the user to define \makelabel for print mode while HTML mode ignores those commands.

⚠ label font When defining \makelabel in a list environment, use \textbf etc. instead of \bfseries.

74.1 List environment

for HTML output:
```
\begin{warpHTML}
```

\LWR@printcloselist May be locally redefined by enumerate or description.

```
\newcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}
```

\LWR@printopenlist May be locally redefined by enumerate or description.

```
\newcommand*{\LWR@printopenlist}{%
  ul style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
}
```

\@mklab Removes PDF spacing.

```
\AtBeginDocument{
\def\@mklab#1{%
%     \hfil %
    #1}
\let\makelabel\@mklab
}
```

\@donoparitem Modified for HTML output by replacing TEX boxes with plain text. Also removes PDF spacing.

```
\def\@donoparitem{%
  \@noparitemfalse
%   \global\setbox\@labels\hbox{\hskip -\leftmargin
%                               \unhbox\@labels
%                                \hskip \leftmargin}%
%   \if@minipage\else
%     \@tempskipa\lastskip
%     \vskip -\lastskip
%     \advance\@tempskipa\@outerparskip
%     \advance\@tempskipa -\parskip
%     \vskip\@tempskipa
%   \fi
}
```

\@item Modified for HTML output by replacing TEX boxes with plain text. Also removes PDF spacing.

```
\def\LWR@HTML@item[#1]{%
\LWR@traceinfo{@item}%
  \if@noparitem
    \@donoparitem
  \else
%     \if@inlabel
%       \indent
%     \fi
    \ifhmode
%       \unskip\unskip
    \fi
    \if@newlist
      \if@nobreak
        \@nbitem
      \else
%         \addpenalty\@beginparpenalty
%         \addvspace\@topsep
%         \addvspace{-\parskip}%
      \fi
    \else
%       \addpenalty\@itempenalty
%       \addvspace\itemsep
    \fi
    \global\@inlabeltrue
  \fi
%   \everypar{%
    \@minipagefalse
    \global\@newlistfalse

%     \if@inlabel
%       \global\@inlabelfalse

%       {\setbox\z@\lastbox
%        \ifvoid\z@
%          \kern-\itemindent
%        \fi}%

%       \box\@labels
%       \penalty\z@
%     \fi

%     \if@nobreak
%       \@nobreakfalse
%       \clubpenalty \@M
%     \else
%       \clubpenalty \@clubpenalty
%       \everypar{}%
%     \fi}%

  \if@noitemarg
    \@noitemargfalse
    \if@nmbrlist

      \refstepcounter\@listctr
    \fi
  \fi

    \makelabel{#1} % extra space
```

```
%   \sbox\@tempboxa{\makelabel{#1}%
%   \global\setbox\@labels\hbox{%
%     \unhbox\@labels
%     \hskip \itemindent
%     \hskip -\labelwidth
%     \hskip -\labelsep
%     \ifdim \wd\@tempboxa >\labelwidth
%       \box\@tempboxa

%     \else
%       \hbox to\labelwidth {\unhbox\@tempboxa}%
%     \fi
%     \hskip \labelsep}%
  \ignorespaces%
}
```

`\@nbitem`

```
\def\@nbitem{%
%   \@tempskipa\@outerparskip
%   \advance\@tempskipa -\parskip
%   \addvspace\@tempskipa
}
```

`\LWR@listitem` [⟨*label*⟩]

Handles `\item` inside a `list`, `itemize`, or `enumerate`.

See `\LWR@openparagraph` where extra `\hspace` is used to leave room for the label while inside a list during paragraph construction.

```
\newcommand*{\LWR@listitem}{%
    \LWR@stoppars%
    \LWR@startnewdepth{listitem}%
    \LWR@htmltag{li}%
    \LWR@orignewline%
    \LWR@startpars%
    \LWR@ensuredoingapar%
    \LWR@origitem%
}
```

`\LWR@nulllistfills` Nullifies various TEX fill commands, in case they are used inside `\makelabel`. Problems are caused when these are nullified all the time.

```
\newcommand*{\LWR@nulllistfills}{%
    \renewcommand*{\hss}{}%
    \renewcommand*{\llap}[1]{##1}%
    \renewcommand*{\rlap}[1]{##1}%
    \renewcommand*{\hfil}{}%
    \renewcommand*{\hfilneg}{}%
    \renewcommand*{\hfill}{}%
}
```

`list` (*env.*) {⟨*label*⟩} {⟨*commands*⟩}

```
\newcommand*{\LWR@liststart}{%
    \LWR@traceinfo{LWR@liststart}%
```

```
    \LWR@stoppars%
    \LWR@pushoneclose{list}%
    \LWR@htmltag{\LWR@printopenlist}\LWR@orignewline%
    \LWR@startpars%
    \setlength{\topsep}{0pt}%
    \setlength{\partopsep}{0pt}%
    \setlength{\itemsep}{0pt}%
    \setlength{\parsep}{0pt}%
    \setlength{\leftmargin}{0pt}%
    \setlength{\rightmargin}{0pt}%
    \setlength{\listparindent}{0pt}%
    \setlength{\itemindent}{0pt}%
    \setlength{\labelsep}{1em}%
    \LWR@nulllistfills%
}

\newcommand*{\LWR@listend}{%
    \LWR@traceinfo{LWR@listend}%
    \LWR@stoppars%
    \LWR@closeprevious{list}%
    \LWR@startpars%
}
```

74.2 Itemize

`\LWR@itemizeitem` [⟨*label*⟩]

Handles `\item` inside an itemize or enumerate.

The optional argument is passed to `\LWR@origitem`.

See `\LWR@openparagraph` where extra `\hspace` is used to leave room for the label while inside a list during paragraph construction.

```
\newcommand*{\LWR@itemizeitem}{%
    \LWR@stoppars%
    \LWR@startnewdepth{listitem}%
    \LWR@htmltag{li}%
    \LWR@orignewline%
    \LWR@startpars%
    \LWR@ensuredoingapar%
    \LWR@origitem%
}
```

`itemize` (*env.*) [⟨*options*⟩]

```
\newcommand*{\LWR@itemizestart}{%
    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
    \renewcommand*{\LWR@printopenlist}{%
      ul style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
    }%
    \LetLtxMacro\item\LWR@itemizeitem%
    \LWR@nulllistfills%
}
```

74.3 Enumerate

An HTML unordered list is used with customized LaTeX-generated labels.

`enumerate` (*env.*) `[`⟨*options*⟩`]`

```
\newcommand*{\LWR@enumeratestart}{%
    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
    \renewcommand*{\LWR@printopenlist}{%
     ul style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
    }%
    \LetLtxMacro\item\LWR@itemizeitem%
    \LWR@nulllistfills%
}
```

74.4 Description

`\LWR@descitem` `[`⟨*label*⟩`]` Handles an `\item` inside a description.

```
\newcommand*{\LWR@descitem}[1][]{%
    \LWR@stoppars%
    \LWR@setlatestname{#1}%
    \LWR@startnewdepth{descitem}%
```

While creating the label, encase it inside tags and disable `\hspace`, which is used by the standard classes to add space to the labels.

```
    \begingroup%
    \let\LWR@orig@desc@makelabel\makelabel
    \renewcommand*{\makelabel}[1]{%
        \LWR@htmltag{dt}%
        \LWR@orig@desc@makelabel{#1}%
        \LWR@htmltag{/dt}%
    }
    \RenewDocumentCommand{\hspace}{s m}{}%
    \LWR@origitem[#1]%
    \endgroup%
    \LWR@orignewline%
    \LWR@htmltag{dd}%
    \LWR@startpars%
}
```

`description` (*env.*) `[`⟨*options*⟩`]`

Footnotes are modified to correctly parse optional arguments.

```
\newcommand*{\LWR@descriptionstart}{%
    \renewcommand*{\LWR@printcloselist}{\LWR@printclosedescription}
    \renewcommand*{\LWR@printopenlist}{dl}
    \LetLtxMacro\item\LWR@descitem%
    \LWR@nulllistfills%
}
```

74.5 Patching the lists

`\LWR@patchlists` Patches list environments.

`\LWR@patchlists` remembers `\item` as defined by whatever packages have been loaded, then patches the `itemize`, `enumerate`, and `description` environments and `\item`. This works with the native LaTeX environments, as well as those provided by enumitem, enumerate, and paralist.

```
\newcommand*{\LWR@patchlists}{%
    \LetLtxMacro\item\LWR@listitem%
    \LetLtxMacro\@item\LWR@HTML@item%
    \renewcommand*{\@trivlist}{%
        \LWR@traceinfo{@trivlist start}%
        \LWR@liststart%
        \LWR@orig@trivlist%
        \LWR@traceinfo{@trivlist done}%
    }%
    \renewcommand*{\trivlist}{%
        \LWR@traceinfo{trivlist}%
        \LWR@origtrivlist%
    }%
    \renewcommand*{\endtrivlist}{%
        \LWR@traceinfo{endtrivlist start}%
        \LWR@origendtrivlist\LWR@listend%
        \LWR@traceinfo{endtrivlist done}%
    }%
    \renewcommand*{\itemize}{%
        \LWR@itemizestart\LWR@origitemize%
    }%
    \renewcommand*{\enumerate}{%
        \LWR@enumeratestart\LWR@origenumerate%
    }%
    \renewcommand*{\description}{%
        \LWR@descriptionstart\LWR@origdescription%
    }%
}
```

`\LWR@restoreoriglists` Restores the original `trivlist` environment.

```
\newcommand*{\LWR@restoreoriglists}{%
    \LWR@traceinfo{LWR@restoreoriglists}%
    \LetLtxMacro\item\LWR@origitem%
    \LetLtxMacro\@item\LWR@orig@item%
    \let\@trivlist\LWR@orig@trivlist%
    \let\trivlist\LWR@origtrivlist%
    \let\endtrivlist\LWR@origendtrivlist%
    \LetLtxMacro\itemize\LWR@origitemize%
    \LetLtxMacro\enditemize\LWR@endorigitemize%
    \LetLtxMacro\enumerate\LWR@origenumerate%
    \LetLtxMacro\endenumerate\LWR@endorigenumerate%
    \LetLtxMacro\description\LWR@origdescription%
    \LetLtxMacro\enddescription\LWR@endorigdescription%
    \let\@mklab\LWR@orig@mklab%
    \let\makelabel\LWR@origmakelabel%
    \let\@donoparitem\LWR@orig@donoparitem%
    \let\@nbitem\LWR@orig@nbitem%
}
```

```
8746 \end{warpHTML}
```

75 Tabular

This is arguably the most complicated part of the entire package. Numerous tricks are employed to handle the syntax of the LaTeX core and the various tabular-related packages.

75.1 Limitations

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, siunitx S columns, or the packages multirow, longtable, supertabular, or xtab.

Defining macros and environments:

⚠ Misplaced alignment tab character &

- When defining environments or macros which include `tabular` and instances of the `&` character, it may be necessary to make `&` active before the environment or macro is defined, then restore `&` to its default catcode after, using the following commands. These are are ignored in print mode.

```
\StartDefiningTabulars
<define macros or environments using tabular and &
here>
\StopDefiningTabulars
```

⚠ floatrow

This includes before and after defining any macro which used `\ttabbox` from floatrow.

⚠ `tabular` inside another environment

- When creating a new environment which contains a `tabular` environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use `\ResumeTabular` as follows. This is ignored in print mode.

```
\StartDefiningTabulars  %    (&  is  used  in  a
definition)
\newenvironment{outerenvironment}
{
  \tabular{cc}
  left & right \\
}
{
  \TabularMacro\ResumeTabular
  left & right \\
  \endtabular
}
\StopDefiningTabulars
```

For developers:

- To automate the use of `\StartDefiningTabulars` and `\EndDefiningTabulars`, these macros may be embedded inside an HTML environment definition to automatically change the catcode of `&` before absorbing the arguments. Another environment may be embedded as well.

```
% Does the work after the catcode has been changed:
\newcommand*{\LWR@HTML@subsomename}[2]{%
  . . .
  \otherenvironmentname  [<args>]  {<args>}  %  for
example
}
% Change catcode before absorbing arguments:
\newcommand*{\LWR@HTML@somename{%
  \StartDefiningTabulars
  \LWR@HTML@subsomename
}
% Change catcode again at the end:
\newcommand*{\LWR@HTML@endsomename}{%
  . . .
  \endotherenvironmentname   % for example
  \StopDefiningTabulars
}
% Combine with the existing print definition:
\LWR@formattedenv{somename}
```

Cell contents:

⚠ macro in a table

- Using a custom macro inside a tabular data cell may result in an extra HTML data cell tag, corrupting the HTML table. To avoid this, use `\TabularMacro` just before the macro. This is ignored in print mode.

```
\TabularMacro\somemacro & more row contents \\
```

Column specifiers:

⚠ math

- Due to the way math is gathered for processing, column specifiers such as `>{$}c<{$}` do not work with lwarp. Instead, each cell must specify math mode individually.

`@` and `!`

- Only one each of `@` and `!` is used at each column, and they are used in that order.

`\multirow`

- In `\multirow` cells, the print version may have extra instances of <, >, `@`, and `!` cells on the second and later rows in the `\multirow` which do not appear in the HTML version.

⚠ `\newcolumntype`

- If `\newcolumntype` does not work for HTML, add a simplified column type using `\HTMLnewcolumntype`.

font and alignment

- lwarp detects each of the following, and sets HTML CSS appropriately:

```
>{\centering\arraybackslash}
>{\raggedright\arraybackslash}
>{\raggedleft\arraybackslash}
>{\itshape}
>{\bfseries}
>{\bfseries\itshape}
```

  These may be used with `\newcolumntype`, such as:

```
\newcolumntype{P}[1]{>{\centering\arraybackslash}p{#1}}
```

Rules:

- Doubled `\hlines`, `\midrules`, and vertical rules are supported.

vertical rules

- Vertical rules next to either side of an `@` or `!` column are displayed on both sides of the column.

width and trim

- Width options are honored. Trim options are converted to rounded top corners. Trim corners are not rounded with `@` or `!` columns, and full-width rules ignore trim. When given an optional width, each cell is styled to create the custom border. Without an optional width, the entire row is given a class to assign the standard border.

combined rules

- If you wish to use `\cmidrule` followed by `\bottomrule`, it may be necessary to use:
  ```
  \cmidrule{2-3} \\[-2ex]
  \bottomrule
  ```
  The optional `-2ex` is ignored in HTML, but improves the visual formatting in the print output.

⚠ `\warpprintonly`
⚠ Misplaced `\noalign`

- For `\toprule` and `\bottomrule`, when combined with a `warpprint` or `warpHTML` environment, if a "Misplaced `\noalign`" error occurs, change
  ```
  This & That \endhead
  ```
  to
  ```
  \warpprintonly{This & That \endhead}
  ```
  and likewise with the other `\end` headings. Keep the `\endfirsthead` row unchanged, as it is still relevent to HTML output.

Other:

- tabularx ignores the width, but `X` columns do produce paragraph columns or multicolumns.

`longtable` headings

- For longtable, place headings and footings which do not apply to HTML inside `\warpprintonly{}`.

⚠ S columns

- For S columns (from the siunitx package), while producing print output, anything non-numeric must be placed inside { } braces, including commands such as `\multirow`. While producing HTML output, though, anything placed inside braces is not seen by lwarp's tabular handling algorithm. To resolve this problem, make a copy of the row, with one version for print output, containing the extra braces, and another version for HTML output, without the extra braces, such as:
  ```
  \warpprintonly{1 & 2 & {\multirow{2}{2cm}{Text}} & 3 \\}
  \warpHTMLonly{1 & 2 & \multirow{2}{2cm}{Text} & 3 \\}
  ```

⚠ `tabular` inside a `<span>`

- In LaTeX, a `tabular` may be placed inside a `minipage`, but in HTML a `<table>` may not be inside a `<span>`. If this situation is detected, a warning is printed instructing the user to isolate the `<span>` using `\warpprintonly` or the `warpprint` environment.

for HTML output: 8747 `\begin{warpHTML}`

75.2 Temporary package-related macros

These macros are temporary placeholders for macros defined by various packages. If the relevent package is not loaded, these placeholders are used instead.

75.2.1 arydshln

Emualated by the original LaTeX non-dashed versions.

```
\LetLtxMacro\hdashline\hline
\LetLtxMacro\cdashline\cline
\LetLtxMacro\firsthdashline\hline
\LetLtxMacro\lasthdashline\hline
```

75.3 Token lookahead

Used by `\LWR@futurenonspacelet` to look at the next token.

`\LWR@mynexttoken`

```
\newcommand\LWR@mynexttoken\relax
```

`\LWR@futurenonspacelet` `\futurelet` copies the next token then executes a function to analyze it.

`\LWR@futurenonspacelet` does the same, but ignores intervening spaces and paragraphs.

Based on the booktabs style:

```
\def\LWR@futurenonspacelet#1{\def\LWR@cs{#1}%
\afterassignment\LWR@fnslone\let\nexttoken= }

\def\LWR@fnslone{\expandafter\futurelet\LWR@cs\LWR@fnsltwo}

\def\LWR@fnsltwo{%
    \expandafter\ifx\LWR@cs\@sptoken%
        \let\next=\LWR@fnslthree%
    \else%
        \expandafter\ifx\LWR@cs\par%
            \let\next=\LWR@fnslthree%
        \else%
            \let\next=\nexttoken%
        \fi%
    \fi\next}

\def\LWR@fnslthree{\afterassignment\LWR@fnslone\let\next= }
```

`\LWR@getmynexttoken` Looks ahead and copies the next token into `\LWR@mynexttoken`.

```
\newcommand*{\LWR@getmynexttoken}{%
    \LWR@traceinfo{LWR@getmynexttoken}%
```

⚠ Nothing must follow this next line:

```
    \LWR@futurenonspacelet\LWR@mynexttoken\LWR@tabledatacolumntag
}
```

75.4 Tabular variables

In order to support nested `tabulars`, each of these is used locally. For local counters, etoolbox's `\defcounter` and lwarp's new `\defaddtocounter` are used.

`LWR@startedrow` (*bool*) True if should print a row tag before this column.

```
\newbool{LWR@startedrow}
\boolfalse{LWR@startedrow}
```

LWR@tabularcelladded (*bool*) True if have added a data cell for this position.

```
\newbool{LWR@tabularcelladded}
\boolfalse{LWR@tabularcelladded}
```

LWR@hlines (*Ctr*) Number of \hlines or \midrules above the next row.

```
\newcounter{LWR@hlines}
```

LWR@hdashedlines (*Ctr*) Number of arydshln dashed lines above the next row.

```
\newcounter{LWR@hdashedlines}
```

LWR@doingtbrule (*bool*) True if the next row will have a top/bottom rule above it.

```
\newbool{LWR@doingtbrule}
\boolfalse{LWR@doingtbrule}
```

LWR@doingcmidrule (*bool*) True if the next row will have a cmidrule above it.

This is used by \LWR@tabularfinishrow to force a final empty row to create the border for the \cmidrule.

```
\newbool{LWR@doingcmidrule}
\boolfalse{LWR@doingcmidrule}
```

LWR@tableparcell (*bool*) True if are handling a paragraph inside a table cell, so must close the paragraph tag before moving on.

```
\newbool{LWR@tableparcell}
```

LWR@skippingmrowcell (*bool*) True if are doing an empty \multirow cell, and thus there is no data tag to close.

```
\newbool{LWR@skippingmrowcell}
```

LWR@skippingmcolrowcell (*bool*) True if are doing an empty \multicolumnrow cell, and thus there is no data tag to close, and do not print @ and ! columns.

```
\newbool{LWR@skippingmcolrowcell}
```

LWR@usedmultirow (*bool*) Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
\newbool{LWR@usedmultirow}
```

LWR@foundmrowcell (*bool*) Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
\newbool{LWR@foundmrowcell}
```

LWR@skipatbang (*bool*) True if just finished a \multicolumn so should not create the trailing @ or ! columns table data cells.

```
\newbool{LWR@skipatbang}
```

LWR@emptyatbang (*bool*) True if finishing a row and should print empty @ or ! column table data cells.

```
\newbool{LWR@emptyatbang}
```

LWR@intabularmetadata (*bool*) True if are in a tabular but not in a data cell. Used to prevent extra HTML breaks if not inside table data.

```
\newbool{LWR@intabularmetadata}
\boolfalse{LWR@intabularmetadata}
```

LWR@exitingtabular (*bool*) When \end is found, turns off the next opening data tag.

```
\newbool{LWR@exitingtabular}
```

LWR@tabularmutemods (*bool*) Mutes HTML output for @, !, < and >.

This is used while printing the final row to generate \bottomrules.

```
\newbool{LWR@tabularmutemods}
```

LWR@tabularfinalrow (*bool*) Used to set aria-hidden if adding a final row for the purpose of adding the bottom border.

```
\newbool{LWR@tabularfinalrow}
```

LWR@validtablecol (*bool*) True if found a valid table column type.

```
\newbool{LWR@validtablecol}
```

LWR@opttablecol (*bool*) True if found a table column optional argument.

```
\newbool{LWR@opttablecol}
```

Used to add a style to a table data cell:

```
\newbool{LWR@tdhavecellstyle}
```

LWR@tabularDepth (*Ctr*) Tracks whether & is being used inside a tabular.

```
\newcounter{LWR@tabulardepth}
\setcounter{LWR@tabulardepth}{0}
```

LWR@tabularpardepth (*Ctr*) Tracks whether should look ahead at the next token when encountering a \par while processing tabular contents.

When LWR@tabularpardepth is deeper than LWR@tabulardepth then lwarp has started looking at the contents of the tabular, and thus any \pars encountered must be followed by another token lookahead.

```
\newcounter{LWR@tabularpardepth}
\setcounter{LWR@tabularpardepth}{0}

\newcommand*{\LWR@colsresult}{}%temp storage for column format results
\newcommand*{\LWR@pposition}{}
\newcommand*{\LWR@pleft}{}
\newcommand*{\LWR@pright}{}
```

`LWR@tablecolspec` Holds the parsed column specification, of total width `LWR@tabletotalLaTeXcols`, not counting `@` and `!` columns.

Will contain a string such as `llrrccpc`, exactly one letter per LaTeX table column, without `@`, `!`, `>`, `<`, or the vertical bar.

`\LWR@strresult` Holds the result of `Str` functions.

```
8807 \providecommand*{\LWR@strresult}{}
8808 \providecommand*{\LWR@strresulttwo}{}
```

`\LWR@origcolspec` Holds the original column specs given to `tabular`.

```
8809 \newcommand*{\LWR@origcolspec}{}
```

`LWR@tablecolspecwidth` (*Ctr*) Holds the number of tokens in the table columns specification.

This is includes one for each `@`, `!`, `<`, `>` column, and also one for each of the parameters of `p`, `@`, `!`, `<`, `>` columns, and three for each `D` column.

(This is not the total # of LaTeX columns in the table.)

```
8810 \newcounter{LWR@tablecolspecwidth}
```

`LWR@tablecolspecindex` (*Ctr*) While parsing the LaTeX table column specification, starts at 1 and is incremented per token of the specification.

```
8811 \newcounter{LWR@tablecolspecindex}
```

`LWR@tableLaTeXcolindex` (*Ctr*) While producing the table, resets to 1 at the start of the table and also at each end of line, and is incremented by 1 by each ampersand.

```
8812 \newcounter{LWR@tableLaTeXcolindex}
```

`LWR@tabletotalLaTeXcols` (*Ctr*) While parsing a table column specification, begins at 0 and increments by 1 per LaTeX table column. Eventually holds the final number of LaTeX table columns in each row, not counting `@` and `!` columns. (In HTML, `@` and `!` cells become their own columns, but are not included in `LWR@tabletotalLaTeXcols`.)

```
8813 \newcounter{LWR@tabletotalLaTeXcols}
```

`LWR@tabletotalLaTeXcolsnext` (*Ctr*) Holds the next LaTeX table column index while parsing, equal to one more than `LWR@tabletotalLaTeXcols`.

```
8814 \newcounter{LWR@tabletotalLaTeXcolsnext}
```

`LWR@colatspec` A data array of specifications for `@` columns. The leftmost's index is `leftedge`, the others are counter values. See section 42.

`LWR@colbangspec` A data array of specifications for `!` columns. The leftmost's index is `leftedge`, the others are counter values. See section 42.

`LWR@colbeforespec` A data array of specifications for `>` columns.

`LWR@colafterspec` A data array of specifications for `<` columns.

LWR@colbarspec A data array of specifications for vertical rules.

LWR@coladdclass A data array of extra css class, as set by >.

LWR@cellcolordepth (*Ctr*) Counts how many cell color s were added to the current tabular data cell.

```
\newcounter{LWR@cellcolordepth}
```

75.4.1 Multicolumn variables

```
\newcounter{LWR@tablemulticolswidth}
```

Indexes into the multicolumn specification:

```
\newcounter{LWR@tablemulticolspos}
```

Remembers multicolumn vertical rules if found in the column spec.

```
\newcounter{LWR@mcolvertbarsl}
\newcounter{LWR@mcolvertbarsr}
\newcounter{LWR@mcolvertbarsldash}
\newcounter{LWR@mcolvertbarsrdash}
\newbool{LWR@mcolvertbaronleft}
```

75.4.2 Longtable variables

LWR@starredlongtable (*bool*) Per the caption package, step the counter if longtable*.

```
\newbool{LWR@starredlongtable}
\boolfalse{LWR@starredlongtable}
```

75.4.3 Midrule variables

LWR@midrulecounter (*Ctr*) Indexes across the LWR@midrules and LWR@trim<l/r>rules data arrays.

```
\newcounter{LWR@midrulecounter}
```

75.5 Handling &, @, !, and bar

For technical discussion regarding problems redefining \&, See:
http://tex.stackexchange.com/questions/11638/
where-do-i-find-futurelets-nasty-behaviour-documented/11860#11860

\LWR@instertatbangcols

```
\newcommand*{\LWR@insertatbangcols}{%
    \ifbool{LWR@skipatbang}%
    {}%
    {%
        \LWR@printatbang{at}{\arabic{LWR@tableLaTeXcolindex}}%
        \LWR@printatbang{bang}{\arabic{LWR@tableLaTeXcolindex}}%
    }%
}
```

\LWR@closetabledatacell If LWR@skippingmrowcell or LWR@skippingmcolrowcell then there is no data tag to close. Otherwise, close any paragraphs, then close the data tag.

```
\newcommand*{\LWR@closetabledatacell}{%
    \booltrue{LWR@intabularmetadata}%
    \ifbool{LWR@exitingtabular}%
    {%

        \LWR@stoppars%
    }%
    {% not exiting tabular
     \ifboolexpr{bool{LWR@skippingmrowcell} or bool{LWR@skippingmcolrowcell}}%
        {%

            \LWR@stoppars%
```

If not skipping a \multicolumnrow cell, insert the @ and ! columns after this non-existant column.

```
            \ifbool{LWR@skippingmcolrowcell}%
                {}%
                {\LWR@insertatbangcols}%
        }%
        {% not skippingmrowcell
```

Insert any < then any @ and ! column contents, unless muted for the \bottomrule or a \multicolumn:

```
            \unskip%
            \ifboolexpr{%
                bool{LWR@tabularmutemods} or
                bool{LWR@skipatbang} or
                bool{LWR@emptyatbang}
            }%
                {}%
                {%
                    \LWR@getexparray{LWR@colafterspec}%
                        {\arabic{LWR@tableLaTeXcolindex}}%
                }%
```

Close paragraphs:

```
            \LWR@stoppars%
            \boolfalse{LWR@tableparcell}%
```

Close the table data cell.

Close any color s.

```
        \whileboolexpr{test {\ifnumcomp{\value{LWR@cellcolordepth}}{>}{0}}}{%
                \LWR@htmltag{/div}\LWR@orignewline%
                \defaddtocounter{LWR@cellcolordepth}{-1}%
            }%
```

Skip the @ and ! cells if are closing a multicolumn cell.

```
            \leavevmode\unskip\LWR@htmltag{/td}\LWR@orignewline%
```

```
            \global\booltrue{LWR@tabularcelladded}%
            \LWR@insertatbangcols%
        }% not skipping mrowcell
    }% not exiting tabular
    \boolfalse{LWR@skippingmrowcell}%
    \boolfalse{LWR@skippingmcolrowcell}%
    \boolfalse{LWR@skipatbang}%
```

Color control. Column is set by >{} for each cell, so it must be cleared here.

```
    \def\LWR@cellHTMLcolor{}%
    \def\LWR@columnHTMLcolor{}%
    \defcounter{LWR@cellcolordepth}{0}%
}
```

When not used inside a tabular, & performs its original function as recorded here (with catcode 4).

```
\let\LWR@origampmacro&
```

```
\end{warpHTML}
```

75.5.1 Handling &

for HTML output:

```
\begin{warpHTML}
```

& Will behave depending on whether it is being used inside tabular.

& is redefined to test whether it is inside a tabular environment, in which case it performs special processing for HTML conversion. If not, it behaves normally.

```
\newcommand*{\LWR@tabularampersand}{%
    \LWR@traceinfo{LWR@tabularampersand}%
    \ifnumcomp{\value{LWR@tabulardepth}}{>}{0}%
    {%
```

If not skipping a multirow cell, close the current data cell.

```
        \unskip%
        \LWR@closetabledatacell%
```

Move to the next column.

```
        \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
        \global\boolfalse{LWR@tabularcelladded}%
```

Look at the next token to decide multi or single column data tag.

```
        \LWR@getmynexttoken%
    }%
```

If not inside a tabular, performs the original action:

```
    {%
        \LWR@origampmacro%
    }%
}
```

& is left with its original catcode for now.

tikz package seems to require & be left alone until after tikz has been loaded. Also, cleveref uses the ampersand in one of its options.

& is made active inside a tabular.

& is left alone when in math alignments.

75.6 Filling an unfinished row

\LWR@tabularfinishrow Adds empty table cells if necessary to finish the row.

At the end of the table, if any bottom rules are requested then an empty row must be generated to form the borders which show the rules.

```
\newcommand*{\LWR@tabularfinishrow}{%
```

If not exiting the tabular, or doing a rule, or have already started a row, finish this row:

```
    \ifboolexpr{%
        not bool {LWR@exitingtabular} or%
        bool{LWR@doingtbrule} or%
        bool{LWR@doingcmidrule} or%
        test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
        test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
        bool{LWR@startedrow}%
    }{%
```

Temporarily turn off LWR@exitingtabular so that table data tags will still be generated.

If generating a final row for the \bottomrule borders, turn off the @, !, <, and > column output:

```
    \ifbool{LWR@exitingtabular}{%
        \booltrue{LWR@tabularmutemods}%
    }{%
        \boolfalse{LWR@tabularmutemods}%
    }%
```

Locally reenable the table data tags until finished with the final row:

```
    \boolfalse{LWR@exitingtabular}%
```

Generate table data tags and ampersands until the right edge:

```
    \whileboolexpr{%
```

```
        test {
            \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
                {\value{LWR@tabletotalLaTeXcols}}
        } or %
        (%
            bool{LWR@intabularmetadata} and%
            not bool{LWR@tabularcelladded} and%
            test {
                \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}
                    {\value{LWR@tabletotalLaTeXcols}}
            }%
        )%
    }%
    {%
        \LWR@tabledatasinglecolumntag%
```

The following is essentially `\LWR@tabularampersand` with `LWR@emptyatbang` added to empty the following cells:

```
        \LWR@closetabledatacell%
        \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
        \global\boolfalse{LWR@tabularcelladded}%
        \booltrue{LWR@emptyatbang}%
```

Starts the next cell:

```
        \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
                {\value{LWR@tabletotalLaTeXcols}}%
            {\LWR@getmynexttoken}%
            {}%
    }%
```

Reenable the original `LWR@exitingtabular` to close the entire table:

```
    \ifbool{LWR@tabularmutemods}{%
        \booltrue{LWR@exitingtabular}%
    }{%
        \boolfalse{LWR@exitingtabular}%
    }%
    \boolfalse{LWR@tabularmutemods}%

    \boolfalse{LWR@emptyatbang}%
    }{}% ifboolexpr
}
```

75.7 Handling \\

Inside tabular, `\\` is redefined to `\LWR@tabularendofline`

Throws away options `\\[dim]` or `\\*`

`\LWR@tabularendofline`

```
\NewDocumentCommand{\LWR@tabularendofline}{s o}{%
```

Finish the row:

```
    \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
            {\value{LWR@tabletotalLaTeXcols}}%
        {\LWR@tabularfinishrow}%
        {\LWR@closetabledatacell}%
    \LWR@htmltag{/tr}\LWR@orignewline%
```

xcolor row color support:

```
    \@rowc@lors%
```

No longer inside a data cell:

```
    \booltrue{LWR@intabularmetadata}%
```

Not yet started a table row:

```
    \boolfalse{LWR@startedrow}%
```

Additional setup:

```
    \defcounter{LWR@hlines}{0}%
    \defcounter{LWR@hdashedlines}{0}%
    \boolfalse{LWR@doingtbrule}%
    \boolfalse{LWR@doingcmidrule}%
    \LWR@clearmidrules%

    \def\LWR@rowHTMLcolor{}%
```

Start at first column:

```
    \defcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
    \global\boolfalse{LWR@tabularcelladded}%
```

Allow TEX to flush the pending paragraph. Not doing so causes a slowdown for very large tables.

```
    \LWR@stoppars%
    \LWR@forceemptyline%
```

Look at the next token to decide between single column data tag or a special case:

```
    \LWR@getmynexttoken%
}
```

75.8 Looking ahead in the column specifications

`\LWR@columnspeclookahead {`⟨*offset*⟩`}`

Looks `offset` tokens ahead in the column specification, setting `\LWR@strresulttwo`.

The `w` column alignment will be seen as a single unit such as `{c}`.

```
\newcommand*{\LWR@columnspeclookahead}[1]{%
    \setcounter{LWR@tempcountone}{\value{LWR@tablecolspecindex}}%
    \addtocounter{LWR@tempcountone}{#1}%
    \fullexpandarg%
   \StrChar{\LWR@origcolspec}{\arabic{LWR@tempcountone}}[\LWR@strresulttwo]%
```

Get the contents of the first group in `\LWR@strresulttwo`:

```
        \exploregroups%
        \StrChar{\LWR@strresulttwo}{1}[\LWR@strresulttwo]%
        \noexploregroups%
}
```

75.9 Parsing @, >, <, !, bar columns

Holds the parsed argument for `@`, `>`, `<`, or `!` columns:

```
\newcommand*{\LWR@colparameter}{}
```

`\LWR@parseatcolumn` {⟨*this column type*⟩}

Handles `@{text}` columns.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

```
\newcommand*{\LWR@parseatcolumn}[1]{%
```

Move to the next token after the '@':

```
    \LWR@traceinfo{at column}%
    \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into `\LWR@colparameter`, expanding once:

```
    \LWR@traceinfo{about to read the next token:}%
    \expandarg%
    \StrChar{\LWR@origcolspec}%
        {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
    \fullexpandarg%
```

Store the result into a data array, expanding once out of `\LWR@colparameter`:

```
    \LWR@traceinfo{have now read the next token}%
    \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
    {% left edge of the table:
        \LWR@traceinfo{at the left edge}%
        \LWR@setexparray{LWR@colatspec}%
            {leftedge}%
            {\expandafter\@firstofone\LWR@colparameter}%
        \LWR@traceinfo{at the left edge: %
            \LWR@getexparray{LWR@colatspec}{leftedge}}%
    }%
    {% not at the left edge:
        \LWR@traceinfo{not at the left edge}%
        \LWR@setexparray{LWR@colatspec}%
            {\arabic{LWR@tabletotalLaTeXcols}}%
```

```
            {\expandafter\@firstofone\LWR@colparameter}%
        \LWR@traceinfo{at \arabic{LWR@tabletotalLaTeXcols}%
        : % space
        \LWR@getexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcols}}}%
    }%
    \let\LWR@colparameter\relax%
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@parsebangcolumn` {⟨*this column type*⟩} Handles `!{text}` columns.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

```
\newcommand*{\LWR@parsebangcolumn}[1]{%
```

Move to the next token after the '!':

```
    \LWR@traceinfo{bang column}%
    \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into `\LWR@colparameter`, expanding once:

```
    \LWR@traceinfo{about to read the next token:}%
    \expandarg%
    \StrChar{\LWR@origcolspec}%
        {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
    \fullexpandarg%
```

Store the result into a data array, expanding once out of `\LWR@colparameter`:

```
    \LWR@traceinfo{have now read the next token}%
    \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
    {% left edge of the table:
        \LWR@traceinfo{at the left edge}%
        \LWR@setexparray{LWR@colbangspec}%
            {leftedge}%
            {\expandafter\@firstofone\LWR@colparameter}%
    }%
    {% not at the left edge:
        \LWR@traceinfo{not at the left edge}%
        \LWR@setexparray{LWR@colbangspec}%
            {\arabic{LWR@tabletotalLaTeXcols}}%
            {\expandafter\@firstofone\LWR@colparameter}%
      \LWR@traceinfo{bang \arabic{LWR@tabletotalLaTeXcols}: \LWR@colparameter!}%
    }%
    \let\LWR@colparameter\relax%
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@checkbeforeaddclass` {⟨*compared csname*⟩} {⟨*css class to add*⟩}

```
\newcommand*{\LWR@checkbeforeaddclass}[2]{%
    \ifcsstrequal{LWR@tempone}{#1}%
        {%
            \LWR@setexparray{LWR@coladdclass}%
                {\arabic{LWR@tabletotalLaTeXcolsnext}}%
                { #2}% space is intentional
        }{}%
```

```
}
```

`\LWR@checkmathcolpar` Error if using math in column parameters.

```
\newcommand*{\LWR@checkmathcolpar}{%
    \IfSubStr{\detokenize\expandafter{\LWR@colparameter}}{\LWRdollar}%
        {%
            \PackageError{lwarp}%
                {%
                Lwarp does not support ‘$’ in column specifiers.\MessageBreak
                  Specify ‘$’ math for each cell in the column.\MessageBreak
                    Enter ’h’ for more info%
                }%
                {%
              For example, replace ‘>{$}c<{$}’ with ‘c’, and then\MessageBreak
                    use ‘$cell contents$’ for each cell in the column.%
                }%
        }{}%
}
```

`\LWR@parsebeforecolumn` {⟨*this column type*⟩}

Handles `>{text}` columns.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

```
\newcommand*{\LWR@parsebeforecolumn}[1]{%
```

Move to the next token after the ’>’:

```
    \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into `\LWR@colparameter`:

```
    \expandarg%
    \StrChar{\LWR@origcolspec}%
        {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
    \fullexpandarg%
```

Error if using `>{$}`, which is not supported by lwarp.

```
    \LWR@checkmathcolpar%
```

Store the result into a data array, expanding once out of `\LWR@colparameter`:

```
    \LWR@setexparray{LWR@colbeforespec}%
        {\arabic{LWR@tabletotalLaTeXcolsnext}}%
        {\expandafter\@firstofone\LWR@colparameter}%
%
    \edef\LWR@tempone{\expandafter\@firstofone\LWR@colparameter}%
```

If detect `>{\centering\arraybackslash}` or related, add a css class.

```
    \LWR@checkbeforeaddclass{LWR@detect@centeringarraybackslash}{tdcenter}
    \LWR@checkbeforeaddclass{LWR@detect@raggedrightarraybackslash}{tdleft}
    \LWR@checkbeforeaddclass{LWR@detect@raggedleftarraybackslash}{tdright}
    \LWR@checkbeforeaddclass{LWR@detect@itshape}{tditshape}
```

```
9070     \LWR@checkbeforeaddclass{LWR@detect@bfseries}{tdbfseries}
9071     \LWR@checkbeforeaddclass{LWR@detect@bfit}{tdbfit}

9072     \let\LWR@colparameter\relax%
9073     \booltrue{LWR@validtablecol}%
9074 }
```

\LWR@parseaftercolumn {⟨*this column type*⟩}

Handles <{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9075 \newcommand*{\LWR@parseaftercolumn}[1]{%
```

Move to the next token after the '<':

```
9076     \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into \LWR@colparameter:

```
9077     \expandarg%
9078     \StrChar{\LWR@origcolspec}%
9079         {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9080     \fullexpandarg%
```

Error if using >{$}, which is not supported by lwarp.

```
9081     \LWR@checkmathcolpar%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
9082     \LWR@setexparray{LWR@colafterspec}%
9083         {\arabic{LWR@tabletotalLaTeXcols}}%
9084         {\expandafter\@firstofone\LWR@colparameter}%
9085     \let\LWR@colparameter\relax%
9086     \booltrue{LWR@validtablecol}%
9087 }
```

\LWR@parsebarcolumn {⟨*this column type*⟩}

Handles vertical rules.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9088 \newcommand*{\LWR@parsebarcolumn}[1]{%
9089     \LWR@traceinfo{LWR@parsebarcolumn}%
```

Remember the bar at this position:

```
9090     \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9091     {% left edge of the table:
9092         \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
9093         \ifdefstring{\LWR@tempone}{tvertbarl}%
9094         {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldouble}}%
9095         {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarl}}%
9096     }%
```

```
    {% not at the left edge:
        \edef\LWR@tempone{%
         \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
        }%
        \ifdefstring{\LWR@tempone}{tvertbarr}%
        {%
            \LWR@setexparray{LWR@colbarspec}%
                {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdouble}%
        }%
        {%
            \LWR@setexparray{LWR@colbarspec}%
                {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarr}%
        }%
    }%
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@parsecoloncolumn` {⟨*this column type*⟩}

Handles vertical rules.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

```
\newcommand*{\LWR@parsecoloncolumn}[1]{%
    \LWR@traceinfo{LWR@parsecoloncolumn}%
```

Remember the bar at this position:

```
    \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
    {% left edge of the table:
        \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
        \ifdefstring{\LWR@tempone}{tvertbarldash}%
        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldoubledash}}%
        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldash}}%
    }%
    {% not at the left edge:
        \edef\LWR@tempone{%
         \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
        }%
        \ifdefstring{\LWR@tempone}{tvertbarrdash}%
        {\LWR@setexparray{LWR@colbarspec}%
            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdoubledash}}%
        {\LWR@setexparray{LWR@colbarspec}%
            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdash}}%
    }%
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@parsesemicoloncolumn` {⟨*this column type*⟩}

Handles vertical rules.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

The arguments to the column type are absorbed by `\LWR@columntype@<char>`, defined by `\LWR@modifycolumntype`.

```
\newcommand*{\LWR@parsesemicoloncolumn}[1]{%
```

Treat ; as a : column:

```
    \LWR@parsecoloncolumn{}%
}
```

75.10 Parsing common column types

`\LWR@parsenormalcolumn` {⟨*this column type*⟩}

Add to the accumulated column specs, advance counters, and pre-clear another column of at, before, and after specs.

`\newcolumntype` definitons use `\LWR@parsenormalcolumn`, so an HTML and print version are given so that they may work inside a `latexmage`.

The arguments to the column type are absorbed by `\LWR@columntype@<char>`, defined by `\LWR@modifycolumntype`.

```
\newcommand*{\LWR@HTML@LWR@parsenormalcolumn}[1]{%
    \defaddtocounter{LWR@tabletotalLaTeXcols}{1}%
    \defaddtocounter{LWR@tabletotalLaTeXcolsnext}{1}%

  \LWR@setexparray{LWR@tablecolspec}{\arabic{LWR@tabletotalLaTeXcols}}{#1}%

    \LWR@traceinfo{normal column \arabic{LWR@tabletotalLaTeXcols}: #1}%
   \LWR@setexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
  \LWR@setexparray{LWR@colbangspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
  \LWR@setexparray{LWR@colbeforespec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
  \LWR@setexparray{LWR@colafterspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
  \LWR@setexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
  \LWR@setexparray{LWR@coladdclass}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
    \booltrue{LWR@validtablecol}%
}

\newcommand*{\LWR@print@LWR@parsenormalcolumn}[1]{}

\LWR@formatted{LWR@parsenormalcolumn}
```

75.11 Parsing 'w' columns

`W` and `w` columns are handled via array with `\HTMLnewcolumntype`.

75.12 Parsing '*' columns

`\LWR@parsestarcolumn` {⟨*this column type*⟩} Star columns should already have been expanded, so this should never be used.

The arguments to the column type are absorbed by `\LWR@columntype@<char>`, defined by `\LWR@modifycolumntype`.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

```
\newcommand*{\LWR@parsestarcolumn}[1]{}
```

Table 13: Tabular baseline

l	p	m	b	r
			bot	
		mid	bot	
l	par	mid	bot	r
	par	mid		
	par			

75.13 Expanding the star column specifications

`\LWR@expandpreamble` {⟨*tabular preamble*⟩}

From array `\@mkpream`.

The resulting expanded preamble is stored in `\the\@temptokena`. Assign as:

```
\edef\destination{\the\@temptokena}
```

```
\newcommand*{\LWR@expandpreamble}[1]{%
    \edef\@tempa{\@temptokena={#1}}%
    \@tempa%
    \@tempswatrue%
    \@whilesw\if@tempswa\fi{%
        \@tempswafalse\the\NC@list%
    }%
}
```

75.14 Parsing the column specifications

⚠ tabular baselines

HTML CSS cannot exactly match the LaTeX concept of a baseline for a table row. Table 13 shows the LaTeX results for various vertical-alignment choices, with the baseline of the first column drawn across all the columns for comparison. See the p column specification in table 14 for details.

Table 14 describes how each kind of column is converted to HTML.

Table 15 shows the various internal macros generated for each column type.

`\LWR@modifycolumntype` {⟨*1: column type letter*⟩} {⟨*2: number args to ignore*⟩} {⟨*3: csname of the cell action*⟩} {⟨*4: csname of the multicolumn print type action*⟩} {⟨*5: csname of the multicolumn print data action*⟩}

Add HTML functionality to an existing print version column type.

```
\newcommand*{\LWR@modifycolumntype}[5]{%
    \LWR@traceinfo{LWR@modifycolumntype !#1!#2!#3!#4!#5!}%
            \LWR@traceinfo{LWR@modifycolumntype #1}%
            \edef\@tempa{%
                \noexpand\csdef{LWR@columntype@#1}{%
                    \noexpand\@nameuse{#3}{#1}%
                    \noexpand\defaddtocounter{LWR@tablecolspecindex}{#2}%
```

Table 14: Tabular HTML column conversions

Each cell is given a CSS class of `td<columntype>`.

l, r, c: Converted to table cells without paragraph tags.
Uses CSS `vertical-align:middle` so that top or bottom-aligned cells may go above or below this cell.

p: Converted to table cells with paragraph tags. Ref: Table 13, LaTeX places the top line of a parbox aligned with the rest of the text line, so CSS `vertical-align:bottom` is used to have the HTML result appear with the paragraph extending below the L, R, C cells at the middle, if possible. This may be confusing as a P cell may not top-align with an L,R,C cell in the HTML conversion, especially in the presence of a B cell, and two P cells side-by-side will be aligned at the bottom instead of the top. Some adjustment of the CSS may be desired, changing `td.tdp`, `td.tdP`, `td.tdprule`, and `td.tdPrule` to `vertical-align: middle`. Another possibility is to change L,R,C, and P to `vertical-align: top` and not worry about the alignment of B and M cells or trying to approximate LaTeX baselines.

m: With paragraph tags, CSS `vertical-align:middle`.

b: With paragraph tags, CSS `vertical-align:top` so that the bottom of the text is closest to the middle of the text line.

w and W: Converted to `l`, `c`, or `r`. No paragraph tags.

P, M, B: Horizontally-centered versions.

S: Treated as 'c'. Ignores optional argument. From the siunitx package.

D: Treated as 'c'. From the dcolumn package.

@, !, >, <: One each, in that order.

|: Vertical rule.

Unknown: Converted to 'l'.

\newcolumntype: Expands to its replacement text.

\HTMLnewcolumntype: Provides simplified replacement text for HTML.

Table 15: HTML column type internal macros

<coltype>: The single-letter column type, such as c or X.

Created by \LWR@modifycolumntype: Used by lwarp to add HTML functionality to each built-in column type.

\LWR@columntype@<coltype>: Handles tabular columns depending on the type. Calls \LWR@parsenormalcolumn or related, then advances \LWR@tablecolspecindex.

\LWR@columntype@mctype@<coltype>: Generates the \multicolumn HTML cell CSS class. Calls \LWR@printmccoltype@normal or related.

\LWR@columntype@mcdata@<coltype>: Generates the \multicolumn HTML cell data. Calls \LWR@printmccoldata@normal or related.

Created by \newcolumntype: From array.

\NC@find@<coltype>: Internally used to parse the column specifier.

\NC@rewrite@<coltype>: Stores the print-mode replacement text.

Created by \HTMLnewcolumntype: From lwarp.

\LWR@print@NC@rewrite@<coltype>: Copied from \NC@rewrite@<type>.

\LWR@HTML@NC@rewrite@<coltype>: Stores the HTML-mode replacement text.

\NC@rewrite@<coltype>: Redefined to use the print or HTML version.

```
                }%
                \noexpand\csdef{LWR@columntype@mctype@#1}{%
                    \noexpand\@nameuse{#4}{#1}%
                }%
                \noexpand\csdef{LWR@columntype@mcdata@#1}{%
                    \noexpand\@nameuse{#5}{#2}%
                }%
            }%
            \@tempa%
    \LWR@traceinfo{LWR@modifycolumntype done}%
}

\LWR@modifycolumntype{l}{0}{LWR@parsenormalcolumn}
    {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}

\LWR@modifycolumntype{c}{0}{LWR@parsenormalcolumn}
    {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}

\LWR@modifycolumntype{r}{0}{LWR@parsenormalcolumn}
    {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}

\LWR@modifycolumntype{@}{0}{LWR@parseatcolumn}
    {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}

\LWR@modifycolumntype{!}{0}{LWR@parsebangcolumn}
    {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}

\LWR@modifycolumntype{>}{0}{LWR@parsebeforecolumn}
```

```
    {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}

\LWR@modifycolumntype{<}{0}{LWR@parseaftercolumn}
    {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}

\LWR@modifycolumntype{|}{0}{LWR@parsebarcolumn}
    {LWR@printmccoltype@vertbar}{LWR@printmccoldata@skip}

\LWR@modifycolumntype{:}{0}{LWR@parsecoloncolumn}
    {LWR@printmccoltype@colon}{LWR@printmccoldata@skip}

\LWR@modifycolumntype{;}{1}{LWR@parsesemicoloncolumn}
    {LWR@printmccoltype@semicolon}{LWR@printmccoldata@skip}

\LWR@modifycolumntype{p}{1}{LWR@parsenormalcolumn}
    {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}

\LWR@modifycolumntype{m}{1}{LWR@parsenormalcolumn}
    {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}

\LWR@modifycolumntype{b}{1}{LWR@parsenormalcolumn}
    {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
```

A star column:

```
\LWR@modifycolumntype{*}{2}{LWR@parsestarcolumn}
    {LWR@printmccoltype@ignore}{LWR@printmccoldata@skip}
```

`\HTMLnewcolumntype` {⟨*col type*⟩} [⟨*num args*⟩] [⟨*optional arg*⟩] {⟨*replacement text*⟩}

A user-level macro to creates an HTML version of the replacement text for the column type.

This is the equivalent to:

```
\newcommand*{\LWR@HTML@NC@rewrite@<columntype>}[<num args>]
      {\NC@find <replacement text>}
\LWR@formatted{NC@rewrite@<columntype>}
```

```
\NewDocumentCommand{\HTMLnewcolumntype}{m O{0} o m}{%
    \IfValueTF{#3}
    {
        \expandafter\newcommand\expandafter*%
            \csname LWR@HTML@NC@rewrite@#1\endcsname[#2][#3]{\NC@find #4}%
        \LWR@formatted{NC@rewrite@#1}%
    }
    {
        \expandafter\newcommand\expandafter*%
            \csname LWR@HTML@NC@rewrite@#1\endcsname[#2]{\NC@find #4}%
        \LWR@formatted{NC@rewrite@#1}%
    }
}

\end{warpHTML}

for PRINT output: \begin{warpprint}
```

```
\NewDocumentCommand{\HTMLnewcolumntype}{m O{0} o m}{}
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@parsetablecols {`⟨*colspecs*⟩`}`

Scans the column specification left to right.

Builds `\LWR@tablecolspec` with the final specification, one LaTeX column per entry. The final number of LaTeX columns in each row is stored in `LWR@tabletotalLaTeXcols`, which is the number of `&` and `\\` in each line, but which does not include `@`, `!`, `<`, `>` specifications in the count.

```
\newcommand*{\LWR@parsetablecols}[1]{%
    \LWR@traceinfo{LWR@parsetablecols}%
```

Remember the original supplied column spec:

```
    \renewcommand*{\LWR@origcolspec}{#1}%
```

Remove spaces:

```
    \expandarg%
    \StrSubstitute{\LWR@origcolspec}{ }{}[\LWR@origcolspec]%
```

Expand any star columns:

```
    \LWR@expandpreamble{\LWR@origcolspec}%
    \edef\LWR@origcolspec{\the\@temptokena}%
```

The parsed column spec data array, `LWR@tablecolspec`, will be overwritten with new values.

Total number of columns found so far. Also pre-initialize the first several columns of specs:

```
    \defcounter{LWR@tabletotalLaTeXcols}{0}%
    \defcounter{LWR@tabletotalLaTeXcolsnext}{1}%
    \LWR@setexparray{LWR@colatspec}{leftedge}{}%
    \LWR@setexparray{LWR@colatspec}{1}{}%
    \LWR@setexparray{LWR@colatspec}{2}{}%
    \LWR@setexparray{LWR@colatspec}{3}{}%
    \LWR@setexparray{LWR@colbangspec}{leftedge}{}%
    \LWR@setexparray{LWR@colbangspec}{1}{}%
    \LWR@setexparray{LWR@colbangspec}{2}{}%
    \LWR@setexparray{LWR@colbangspec}{3}{}%
    \LWR@setexparray{LWR@colbeforespec}{1}{}%
    \LWR@setexparray{LWR@colbeforespec}{2}{}%
    \LWR@setexparray{LWR@colbeforespec}{3}{}%
    \LWR@setexparray{LWR@colafterspec}{1}{}%
    \LWR@setexparray{LWR@colafterspec}{2}{}%
    \LWR@setexparray{LWR@colafterspec}{3}{}%
    \LWR@setexparray{LWR@colbarspec}{leftedge}{}%
    \LWR@setexparray{LWR@colbarspec}{1}{}%
    \LWR@setexparray{LWR@colbarspec}{2}{}%
```

```
    \LWR@setexparray{LWR@colbarspec}{3}{}%
    \LWR@setexparray{LWR@coladdclass}{1}{}%
    \LWR@setexparray{LWR@coladdclass}{2}{}%
    \LWR@setexparray{LWR@coladdclass}{3}{}%
```

Starting at the first column specification:

```
    \defcounter{LWR@tablecolspecindex}{1}%
```

Place the colspecs string length into `\LWR@strresult`, and remember the number of characters in the column specification:

```
    \expandarg%
    \StrLen{\LWR@origcolspec}[\LWR@strresult]%
    \fullexpandarg%
    \LWR@traceinfo{original column spec length: \LWR@strresult}%
    \defcounter{LWR@tablecolspecwidth}{\LWR@strresult}%
```

Haven't seen any optional arguments so far

```
    \boolfalse{LWR@opttablecol}%
```

Scan through the column specifications:

```
    \whileboolexpr{%
        not test{%
            \ifnumcomp{\value{LWR@tablecolspecindex}}{>}%
                {\value{LWR@tablecolspecwidth}}%
        }%
    }%
    {%
```

Place the next single-character column type into `\LWR@strresult`:

```
    \expandarg%
   \StrChar{\LWR@origcolspec}{\arabic{LWR@tablecolspecindex}}[\LWR@strresult]%
    \LWR@traceinfo{position \arabic{LWR@tablecolspecindex}: \LWR@strresult}%
    \fullexpandarg%
```

Not yet found a valid column type:

```
    \boolfalse{LWR@validtablecol}%
```

Skip over any optional arguments, such as siunitx `S` column:

```
    \IfStrEq{\LWR@strresult}{[}{\booltrue{LWR@opttablecol}}{}%
```

Throw away anything found inside the optional argument:

```
    \ifbool{LWR@opttablecol}%
    {}% inside an optional argument
    {% not an optional tabular argument
```

Not inside an optional argument, so consider the column type:

```
        \ifcsdef{LWR@columntype@\LWR@strresult}%
            {\csuse{LWR@columntype@\LWR@strresult}}%
            {}%
```

If an unknown column type, use `l`:

```
        \ifbool{LWR@validtablecol}{}{%
            \LWR@traceinfo{invalid column type: \LWR@strresult}%
            \LWR@parsenormalcolumn{l}%
        }%
    }% not an optional column argument
```

If read the closing bracket, no longer inside the optional argument:

```
    \IfStrEq{\LWR@strresult}{]}{\boolfalse{LWR@opttablecol}}{}%
```

Move to the next character:

```
    \defaddtocounter{LWR@tablecolspecindex}{1}%
    }% whiledo
}%
```

75.15 colortbl and xcolor tabular color support

These macros provide a minimal emulation of some colortbl macros which might appear between table cells. If colortbl is loaded, these macros will be replaced with functional versions.

For each of the HTML colors below, the text for the HTML color is set if requested, but the macro is empty if none has been set.

`\rownum` Reserve a counter register.

```
\@ifundefined{rownum}{\newcount\rownum}{}
```

`\@rowcolors` Emulated in case xcolor is not used.

```
\newcommand*{\@rowcolors}{}
```

`\@rowc@lors` Emulated in case xcolor is not used.

```
\newcommand*{\@rowc@lors}{}
```

`\LWR@xcolorrowHTMLcolor` Emulated xcolor row color.

```
\newcommand*{\LWR@xcolorrowHTMLcolor}{}
```

`\LWR@columnHTMLcolor` HTMLstyle code for the column color.

```
\def\LWR@columnHTMLcolor{}
```

`\LWR@rowHTMLcolor` HTMLstyle code for the row color.

```
\def\LWR@rowHTMLcolor{}
```

\LWR@cellHTMLcolor HTMLstyle code for the cell color.

```
9308 \def\LWR@cellHTMLcolor{}
```

\LWR@ruleHTMLcolor HTMLstyle code for the rule color.

```
9309 \newcommand*{\LWR@ruleHTMLcolor}{}
```

\rowcolor [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩] Print version. The HTML version is in lwarp-colortbl. Used before starting a tabular data cell, thus \LWR@getmynexttoken.

```
9310 \newcommand*{\rowcolor}{\LWR@getmynexttoken}%
```

\arrayrulecolor [⟨*model*⟩] {⟨*color*⟩}

\arrayrulecolornexttoken [⟨*model*⟩] {⟨*color*⟩}

Print versions for use outside and inside a tabular:

```
9311 \newcommand{\arrayrulecolor}[2][named]{}
9312 \newcommand{\arrayrulecolornexttoken}[2][named]{\LWR@getmynexttoken}
```

\doublerulesepcolor [⟨*model*⟩] {⟨*color*⟩}

\doublerulesepcolornexttoken [⟨*model*⟩] {⟨*color*⟩}

Print versions for use inside and outside a tabular:

```
9313 \newcommand{\doublerulesepcolor}[2][named]{}
9314 \newcommand{\doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}
```

75.16 Starting a new row

\LWR@maybenewtablerow If have not yet started a new table row, begin one now. Creates a new row tag, adding a class for hline or tbrule if necessary.

```
9315 \newcommand*{\LWR@maybenewtablerow}
9316 {%
9317     \ifbool{LWR@startedrow}%
9318     {}% started the row
9319     {% not started the row
```

Pre-compute the aria-hidden attribute, used to hide from screen readers the final row if it is only used to create the bottom border:

```
9320         \ifbool{LWR@tabularfinalrow}%
9321             {%
9322                 \renewcommand*{\LWR@tempone}%
9323                     { aria-hidden=\textquotedbl{}true\textquotedbl}%
9324             }%
9325             {%
9326                 \renewcommand*{\LWR@tempone}{}%
9327             }%
```

Start a new row if doing \hline:

```
        \ifboolexpr{%
            test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
            test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}}%
        }%
        {%
            \LWR@htmltag{%
                tr %
                class=\textquotedbl{}hline\textquotedbl%
                \LWR@tempone% aria-hidden
            }%
            \LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
            \booltrue{LWR@startedrow}%
            \booltrue{LWR@intabularmetadata}%
        }%
```

If not doing \hline, start a row if doing a top or bottom rule:

```
        {% not doing hline
            \ifbool{LWR@doingtbrule}%
            {%
                \ifdefvoid{\LWR@ruleHTMLcolor}{%
                    \LWR@htmltag{%
                        tr %
                        class=\textquotedbl{}tbrule\textquotedbl%
                        \LWR@tempone% aria-hidden
                    }%
                }{%
                    \LWR@htmltag{%
                       tr class=\textquotedbl{}tbrule\textquotedbl\ % space
                        style=\textquotedbl{}border-top: 1px solid % space
                           \LWR@origpound\LWR@ruleHTMLcolor \textquotedbl{}%
                        \LWR@tempone% aria-hidden
                    }%
                }%
                \LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
                \booltrue{LWR@startedrow}%
                \booltrue{LWR@intabularmetadata}%
            }%
            {%
```

If not the final row, start a new row:

```
                \ifbool{LWR@tabularfinalrow}%
                    {}%
                    {%
                        \LWR@htmltag{tr}\LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
                        \booltrue{LWR@startedrow}%
                        \booltrue{LWR@intabularmetadata}%
                    }%
            }%
        }% end of not doing hline
    }% end of not started the row
}
```

75.17 Printing vertical bar tags

`\LWR@printbartag {⟨index⟩}`

Adds to a tabular data cell an HTML class name for a left/right vertical bar.

```
\newcommand*{\LWR@printbartag}[1]{%
    \LWR@traceinfo{LWR@printbartag !#1!}%
    \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
    {}% muting or empty
    {% not muting
        \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{#1}}%
        \ifdefempty{\LWR@tempone}{}{ \LWR@tempone}%
    }% not muting
    \LWR@traceinfo{LWR@printbartag done}%
}
```

75.18 Printing @ or ! tags

`\LWR@printatbang {⟨at — or — bang⟩} {⟨index⟩}`

```
\newcommand*{\LWR@printatbang}[2]{%
```

Fetch the column at or bang spec:

```
    \xdef\LWR@atbangspec{\LWR@getexparray{LWR@col#1spec}{#2}}%
    \LWR@traceinfo{atbang: #2 !\LWR@atbangspec!}%
```

Only generate if is not empty;

```
    \ifdefempty{\LWR@atbangspec}%
    {}%
    {% not empty
        \LWR@htmltag{%
            td class=\textquotedbl{}td#1%
            \LWR@subaddcmidruletrim{}{}%
            \LWR@printbartag{#2}%
            \textquotedbl{}%
            \LWR@tdstartstyles%
            \LWR@addcmidrulewidth%
            \LWR@addcdashline%
            \LWR@addtabularrulecolors%
            \LWR@tdendstyles%
        }%
```

Create an empty cell if muting for the \bottomrule:

```
        \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
            {}%
            {\LWR@atbangspec}%
%
        \LWR@htmltag{/td}\LWR@orignewline%
        \global\booltrue{LWR@tabularcelladded}%
    }% not empty
}%
```

\LWR@addleftmostbartag

```
\newcommand*{\LWR@addleftmostbartag}{%
    \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}{%
        \LWR@printbartag{leftedge}%
    }{}%
}
```

\LWR@tabularleftedge

```
\newcommand*{\LWR@tabularleftedge}{%
    \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}%
    {%
        \LWR@printatbang{at}{leftedge}%
        \LWR@printatbang{bang}{leftedge}%
    }% left edge
    {}% not left edge
}
```

75.19 Cell opening tag

\LWR@thiscolspec Temporary storage.

```
\newcommand*{\LWR@thiscolspec}{}
```

\LWR@tabledatasinglecolumntag Print a table data opening tag with style for alignment and color.

```
\newcommand*{\LWR@tabledatasinglecolumntag}%
{%
    \LWR@traceinfo{LWR@tabledatasinglecolumntag}%
    \LWR@maybenewtablerow%
```

Don't start a new paragraph tag if have already started one:

```
    \ifbool{LWR@intabularmetadata}%
    {%
```

If have found the end of tabular command, do not create the next data cell:

```
        \ifbool{LWR@exitingtabular}{}%
        {% not exiting tabular
```

Print the @ and ! contents before first column:

```
            \LWR@tabularleftedge%
```

Fetch the current column's alignment character into `\LWR@strresult`:

```
            \xdef\LWR@strresult{%
          \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}%
            }%
```

Print the start of a new table data cell:

```
        \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to print td tag}%
           \LWR@htmltag{%
               td class=\textquotedbl{}td%
```

Append this column's spec:

```
               \LWR@strresult%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag. Also add vertical bar tags.

```
               \LWR@addcmidruletrim%
               \LWR@addleftmostbartag%
               \LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
```

Add any tabular > column text alignment or font control CSS:

```
               \LWR@getexparray{LWR@coladdclass}%
                   {\arabic{LWR@tableLaTeXcolindex}}%
```

Close the class description:

```
               \textquotedbl{}%
```

Add styles for rules, alignment:

```
               \LWR@tdstartstyles%
               \LWR@addcmidrulewidth%
               \LWR@addcdashline%

               \xdef\LWR@thiscolspec{%
                   \LWR@getexparray{LWR@tablecolspec}%
                       {\arabic{LWR@tableLaTeXcolindex}}%
               }%
               \LWR@addformatwpalignment{\LWR@thiscolspec}%
```

Add styles for cell and rule colors:

```
               \LWR@addtabularrowcolor%
               \LWR@addtabularrulecolors%

               \LWR@tdendstyles%
           }% HTML td
         \LWR@traceinfo{LWR@tabledatasinglecolumntag: done printing td tag}%
```

If this is a `p`, `m`, `b`, or `X` column, allow paragraphs:

```
           \ifboolexpr{%
               test{ \ifdefstring{\LWR@strresult}{p} } or
```

```
                test{ \ifdefstring{\LWR@strresult}{m} } or
                test{ \ifdefstring{\LWR@strresult}{b} }
            }%
            {% allow pars
           \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to LWR@startpars}%
                \booltrue{LWR@tableparcell}%
                \LWR@startpars%
           \LWR@traceinfo{LWR@tabledatasinglecolumntag: done with LWR@startpars}%
            }% allow pars
            {}% no pars
```

Print the > contents unless muted for the `\bottomrule`:

```
            \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
            {}%
            {%
           \LWR@getexparray{LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
            }%
            \boolfalse{LWR@intabularmetadata}%
        }% not exiting tabular
    }{}% in tabular metadata
    \LWR@traceinfo{LWR@tabledatasinglecolumntag: done}%
}%
```

75.20 Midrules

`LWR@midrules` `LWR@midrules` is a data array (section 42) of columns each containing a non-zero width if a midrule should be created for this column.

`LWR@trimlrules` `LWR@trimlrules` is a data array (section 42) of columns containing `l` if a midrule should be left trimmed for each column.

`LWR@trimrrules` `LWR@trimrrules` is a data array (section 42) of columns containing `r` if a midrule should be right trimmed for each column.

`LWR@cdashlines` `LWR@cdashlines` is a data array (section 42) of columns each containing a `Y` if an arydshln package "cdashed line" should be created for this column.

`\LWR@heavyrulewidth` (*Len*) The default width of the rule.

```
\newlength{\LWR@heavyrulewidth}
\setlength{\LWR@heavyrulewidth}{.08em}
```

`\LWR@lightrulewidth` (*Len*) The default width of the rule.

```
\newlength{\LWR@lightrulewidth}
\setlength{\LWR@lightrulewidth}{.05em}
```

`\LWR@cmidrulewidth` (*Len*) The default width of the rule.

```
\newlength{\LWR@cmidrulewidth}
\setlength{\LWR@cmidrulewidth}{.03em}
```

`\LWR@thiscmidrulewidth` (*Len*) The width of the next rule, defaulting to `\LWR@cmidrulewidth`.

If not `\LWR@cmidrulewidth`, a style will be used to generate the custom width.

Assigned from the LWR@midrules array.

```
\newlength{\LWR@thiscmidrulewidth}
\setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}
```

\LWR@clearmidrules Start new midrules. Called at beginning of tabular and also at \\.

Clears all LWR@midrules and LWR@trimrules markers for this line.

```
\newcommand*{\LWR@clearmidrules}
{%
    \defcounter{LWR@midrulecounter}{1}%
    \whileboolexpr{%
        not test{%
            \ifnumcomp{\value{LWR@midrulecounter}}{>}%
                {\value{LWR@tabletotalLaTeXcols}}%
        }%
    }%
    {%
        \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{0pt}%
        \setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
        \LWR@setexparray{LWR@trimlrules}{\arabic{LWR@midrulecounter}}{}%
        \LWR@setexparray{LWR@trimrrules}{\arabic{LWR@midrulecounter}}{}%
        \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{N}%
        \defaddtocounter{LWR@midrulecounter}{1}%
    }%
}
```

\LWR@subcmidrule {⟨*width*⟩} {⟨*trim*⟩} {⟨*leftcolumn*⟩} {⟨*rightcolumn*⟩}

Marks LWR@midrules data array elements to be non-zero widths from left to right columns. Also marks trimming for the L and/or R columns.

LWR@doingcmidrule is set to force an empty row at the end of the tabular to create the rule.

```
\newcommand*{\LWR@subcmidrule}[4]{%
    \defcounter{LWR@midrulecounter}{#3}%
    \whileboolexpr{%
        not test {%
            \ifnumcomp{\value{LWR@midrulecounter}}{>}{#4}%
        }%
    }%
    {%
        \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{#1}%
        \defaddtocounter{LWR@midrulecounter}{1}%
    }% whiledo
    \IfSubStr{#2}{l}{\LWR@setexparray{LWR@trimlrules}{#3}{l}}{}%
    \IfSubStr{#2}{r}{\LWR@setexparray{LWR@trimrrules}{#4}{r}}{}%
    \booltrue{LWR@doingcmidrule}%
}
```

\LWR@docmidrule [⟨*width*⟩] (⟨*trim*⟩) {⟨*leftcolumn-rightcolumn*⟩}

Marks LWR@midrules array elements to be a non-zero width from left to right columns. Also marks trimming for the L and/or R columns.

```
\NewDocumentCommand{\LWR@docmidrule}
    {O{\LWR@cmidrulewidth} D(){} >{\SplitArgument{1}{-}}m}
    {\LWR@subcmidrule{#1}{#2}#3}
```

`\LWR@subcdashline` {⟨*leftcolumn*⟩} {⟨*rightcolumn*⟩}

Marks `LWR@cdashlines` data array elements to be `Y` from left to right columns.

`LWR@doingcmidrule` is set to force an empty row at the end of the tabular to create the rule.

```
\newcommand*{\LWR@subcdashline}[2]{%
    \defcounter{LWR@midrulecounter}{#1}%
    \whileboolexpr{%
        not test {%
            \ifnumcomp{\value{LWR@midrulecounter}}{>}{#2}%
        }%
    }%
    {%
        \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{Y}%
        \defaddtocounter{LWR@midrulecounter}{1}%
    }% whiledo
    \booltrue{LWR@doingcmidrule}%
}
```

`\LWR@docdashline` {⟨*leftcolumn-rightcolumn*⟩}

Marks `LWR@cdashlines` data array elements to be `Y` from left to right columns.

```
\NewDocumentCommand{\LWR@docdashline}{>{\SplitArgument{1}{-}}m}%
{%
    \LWR@subcdashline#1%
}
```

`\LWR@tdstartstyles` Begins possibly adding a table data cell style.

```
\newcommand*{\LWR@tdstartstyles}{\boolfalse{LWR@tdhavecellstyle}}
```

`\LWR@tdaddstyle` Starts adding a table data cell style.

```
\newcommand*{\LWR@tdaddstyle}{%
    \ifbool{LWR@tdhavecellstyle}%
        {; }%
        { style=\textquotedbl}%
    \booltrue{LWR@tdhavecellstyle}%
}
```

`\LWR@tdendstyles` Finishes possibly adding a table data cell style. Prints the closing quote.

```
\newcommand*{\LWR@tdendstyles}{%
    \ifbool{LWR@tdhavecellstyle}%
        {%
            \textquotedbl%
            \boolfalse{LWR@tdhavecellstyle}%
        }{}%
}
```

\LWR@subaddcmidruletrim {⟨*lefttrim*⟩} {⟨*righttrim*⟩} Adds a \cmidrule with optional trim.

```
\newcommand*{\LWR@subaddcmidruletrim}[2]{%
    \setlength{\LWR@templengthone}{%
        \LWR@getexparray{LWR@midrules}{\arabic{LWR@tableLaTeXcolindex}}%
    }%
    \ifdimcomp{\LWR@templengthone}{>}{0pt}%
        {%
```

Print the class with left and right trim letters appended:

```
            \LWR@origtilde tdrule#1#2%
```

Remember the width of the rule:

```
            \setlength{\LWR@thiscmidrulewidth}{\LWR@templengthone}%
        }%
        {%
            \setlength{\LWR@thiscmidrulewidth}{0pt}%
        }%
}
```

\LWR@addcmidruletrim Adds left or right trim to a \cmidrule.

```
\newcommand*{\LWR@addcmidruletrim}{%
    \LWR@subaddcmidruletrim%
       {\LWR@getexparray{LWR@trimlrules}{\arabic{LWR@tableLaTeXcolindex}}}%
       {\LWR@getexparray{LWR@trimrrules}{\arabic{LWR@tableLaTeXcolindex}}}%
}
```

\LWR@addrulewidth {⟨*thiswidth*⟩} {⟨*defaultwidth*⟩}

If not default width, add a custom style with width and color depending on thiswidth.

Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
\newcommand{\LWR@addrulewidth}[2]{%
```

Only add a custom width if thiswidth is different than the defaultwidth, or if a color is being used:

```
    \ifboolexpr{%
        test{\ifdimcomp{#1}{=}{0pt}} or
        (
            ( test{\ifdimcomp{#1}{=}{#2}} and not bool{FormatWP} )
            and ( test {\ifdefvoid{\LWR@ruleHTMLcolor}} )
        )
    }%
    {}% default width and color
    {% custom width and/or color
```

Ensure that the width is wide enough to display in the browser:

```
        \LWR@forceminwidth{#1}%
```

Begin adding another style:

```
        \LWR@tdaddstyle%
```

The style itself:

```
        border-top:\LWR@printlength{\LWR@atleastonept} solid % space
```

If default gray, the darkness of the color depends on the thickness of the rule:

```
        \ifdefvoid{\LWR@ruleHTMLcolor}{%
            \ifdimcomp{#1}{<}{\LWR@lightrulewidth}%
            {\LWR@origpound{}A0A0A0}%
            {% lightrule or heaver
                \ifdimcomp{#1}{<}{\LWR@heavyrulewidth}%
                {\LWR@origpound{}808080}%
                {black}%
            }% lightrule or heavier
        }{%
            \LWR@origpound\LWR@ruleHTMLcolor%
        }%
    }% custom width and/or color
}
```

`\LWR@addcmidrulewidth` Adds a `style` for the rule width.

Must be placed between `\LWR@tdstartstyles` and `\LWR@tdendstyles`.

```
\newcommand{\LWR@addcmidrulewidth}{%
    \LWR@addrulewidth{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
}
```

`\LWR@addcdashline` Must be placed between `\LWR@tdstartstyles` and `\LWR@tdendstyles`.

```
\newcommand{\LWR@addcdashline}{%
    \edef\LWR@tempone{%
        \LWR@getexparray{LWR@cdashlines}{\arabic{LWR@tableLaTeXcolindex}}%
    }%
    \ifdefstring{\LWR@tempone}{Y}{%
        \LWR@tdaddstyle%
        border-top: 1pt dashed %
        \ifdefvoid{\LWR@ruleHTMLcolor}%
            {black}%
            {\LWR@origpound\LWR@ruleHTMLcolor}%
    }{}%
}
```

`\LWR@WPcell` {⟨*text-align*⟩} {⟨*vertical-align*⟩}

```
\newcommand*{\LWR@WPcell}[2]{%
    \LWR@tdaddstyle%
    \LWR@print@mbox{text-align:#1}; \LWR@print@mbox{vertical-align:#2}%
}
```

`\LWR@addformatwpalignment` {⟨*colspec*⟩}

If `FormatWP`, adds a `style` for the alignment.

Must be placed between `\LWR@tdstartstyles` and `\LWR@tdendstyles`.

```
\newcommand*{\LWR@addformatwpalignment}[1]{%
    \ifbool{FormatWP}{%
        \IfSubStr{#1}{l}{\LWR@WPcell{left}{middle}}{}%
        \IfSubStr{#1}{c}{\LWR@WPcell{center}{middle}}{}%
        \IfSubStr{#1}{r}{\LWR@WPcell{right}{middle}}{}%
        \IfSubStr{#1}{p}{\LWR@WPcell{left}{bottom}}{}%
        \IfSubStr{#1}{m}{\LWR@WPcell{left}{middle}}{}%
        \IfSubStr{#1}{b}{\LWR@WPcell{left}{top}}{}%
    }{}%
}
```

75.21 Cell colors

`\LWR@addtabularrowcolor` Adds a cell's row color style, if needed.

No color is added for the final row of empty cells which finishes each `tabular`.

```
\newcommand*{\LWR@addtabularrowcolor}{%
    \ifbool{LWR@tabularmutemods}{}{%
        \ifdefvoid{\LWR@rowHTMLcolor}{%
            \ifdefvoid{\LWR@xcolorrowHTMLcolor}{}%
            {% xcolor row color
                \LWR@tdaddstyle%
                background:\LWR@origpound\LWR@xcolorrowHTMLcolor%
            }%
        }%
        {% explicit row color
            \LWR@tdaddstyle%
            background:\LWR@origpound\LWR@rowHTMLcolor%
        }%
    }%
}
```

`\LWR@addtabularhrulecolor` Adds a cell's horizontal rule color style, if needed.

```
\newcommand*{\LWR@addtabularhrulecolor}{%
```

If either form of horizontal rule is requested:

```
    \ifboolexpr{%
        test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
        test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
        bool{LWR@doingtbrule}%
    }{%
```

If there is a no custom color:

```
        \ifdefvoid{\LWR@ruleHTMLcolor}%
        {%
            \ifnumcomp{\value{LWR@hlines}}{>}{1}%
            {%
                \LWR@tdaddstyle%
                border-top: 4px double%
            }{% else
```

```
            \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
            {%
                \LWR@tdaddstyle%
                border-top: 2px dashed%
            }{% else
            \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
            {%
                \LWR@tdaddstyle%
                border-top: 1px dashed%
            }{}}}%
```

If no color and not doubled or dashed, then add nothing, since a simpler rule is the default.

```
        }%
```

If there is a custom color:

```
        {%
            \ifnumcomp{\value{LWR@hlines}}{>}{1}%
            {%
                \LWR@tdaddstyle%
                border-top: 4px double \LWR@origpound\LWR@ruleHTMLcolor%
            }{% else
            \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
            {%
                \LWR@tdaddstyle%
                border-top: 2px dashed \LWR@origpound\LWR@ruleHTMLcolor%
            }{% else
            \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
            {%
                \LWR@tdaddstyle%
                border-top: 1px dashed \LWR@origpound\LWR@ruleHTMLcolor%
            }{% else
                \LWR@tdaddstyle%
                border-top: 1px solid \LWR@origpound\LWR@ruleHTMLcolor%
            }}}%
        }%
    }{}%
}
```

`\LWR@addtabularrulecolors` Adds a cell's rule color styles, if needed.

No color is added for the final row of empty cells which finishes each `tabular`.

```
\newcommand*{\LWR@addtabularrulecolors}{%
```

Custom horizonal rule color:

```
    \LWR@addtabularhrulecolor%
```

No vertical rules if finishing the `tabular` with a row of empty cells:

```
    \ifbool{LWR@tabularmutemods}{}{%
```

If at the leftmost cell, possibly add a leftmost vertical rule:

```
        \ifnumequal{\value{LWR@tableLaTeXcolindex}}{1}{%
```

Fetch the left edge's vertical bar specification:

```
            \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
```

Add a custom style if a vertical bar was requested:

```
            \ifdefstring{\LWR@tempone}{tvertbarl}{%
                    \LWR@tdaddstyle%
                    border-left: 1px solid % space
                        \LWR@vertruleHTMLcolor%
            }{}%
            \ifdefstring{\LWR@tempone}{tvertbarldouble}{%
                    \LWR@tdaddstyle%
                    border-left: 4px double % space
                        \LWR@vertruleHTMLcolor%
            }{}%
            \ifdefstring{\LWR@tempone}{tvertbarldash}{%
                    \LWR@tdaddstyle%
                    border-left: 1px dashed % space
                        \LWR@vertruleHTMLcolor%
            }{}%
            \ifdefstring{\LWR@tempone}{tvertbarldoubledash}{%
                    \LWR@tdaddstyle%
                    border-left: 2px dashed % space
                        \LWR@vertruleHTMLcolor%
            }{}%
        }{}%
```

Possibly add a right vertical rule for this cell:

```
        \edef\LWR@tempone{%
          \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tableLaTeXcolindex}}%
        }%
        \ifdefstring{\LWR@tempone}{tvertbarr}{%
```

Add a custom style if a vertical bar was requested:

```
                \LWR@tdaddstyle%
                border-right: 1px solid \LWR@vertruleHTMLcolor%
        }{}%
        \ifdefstring{\LWR@tempone}{tvertbarrdouble}{%
                \LWR@tdaddstyle%
                border-right: 4px double \LWR@vertruleHTMLcolor%
        }{}%
        \ifdefstring{\LWR@tempone}{tvertbarrdash}{%
                \LWR@tdaddstyle%
                border-right: 1px dashed \LWR@vertruleHTMLcolor%
        }{}%
        \ifdefstring{\LWR@tempone}{tvertbarrdoubledash}{%
                \LWR@tdaddstyle%
                border-right: 2px dashed \LWR@vertruleHTMLcolor%
        }{}%
    }%
}
```

`\LWR@subaddtabularcellcolor` {⟨*html color*⟩}

```
\newcommand*{\LWR@subaddtabularcellcolor}[1]{%
    \LWR@htmltag{div class=\textquotedbl{}cellcolor\textquotedbl\ % space
```

```
        style=\textquotedbl{}%
        background:\LWR@origpound{}{}#1 %
        \textquotedbl\ %
    }% space
    \defaddtocounter{LWR@cellcolordepth}{1}%
}
```

`\LWR@addtabularcellcolor` Adds a cell color style, if needed.

```
\newcommand*{\LWR@addtabularcellcolor}{%
    \ifdefvoid{\LWR@cellHTMLcolor}%
    {%
        \ifdefvoid{\LWR@rowHTMLcolor}%
        {%
            \ifdefvoid{\LWR@xcolorrowHTMLcolor}%
            {%
                \ifdefvoid{\LWR@columnHTMLcolor}%
                {}%
                {\LWR@subaddtabularcellcolor{\LWR@columnHTMLcolor}}%
            }%
            {\LWR@subaddtabularcellcolor{\LWR@xcolorrowHTMLcolor}}%
        }%
        {\LWR@subaddtabularcellcolor{\LWR@rowHTMLcolor}}%
    }%
    {\LWR@subaddtabularcellcolor{\LWR@cellHTMLcolor}}%
}
```

75.22 Multicolumns

75.22.1 Parsing multicolumns

`\LWR@printmccoltype@normal` {⟨*col type*⟩}

Prints the column type, and remembers that any vertical bars are no longer on the left edge.

```
\newcommand*{\LWR@printmccoltype@normal}[1]{%
    #1%
    \boolfalse{LWR@mcolvertbaronleft}%
}
```

`\LWR@printmccoltype@ignore` {⟨*col type*⟩}

This type does not print a multi-column data cell.

```
\newcommand*{\LWR@printmccoltype@ignore}[1]{}%
```

`\LWR@printmccoltype@vertbar` {⟨*col type*⟩}

Adds a left or right vertical bar.

```
\newcommand*{\LWR@printmccoltype@vertbar}[1]{%
    \ifbool{LWR@mcolvertbaronleft}%
        {\defaddtocounter{LWR@mcolvertbarsl}{1}}% left edge
        {\defaddtocounter{LWR@mcolvertbarsr}{1}}% not left edge
}
```

`\LWR@printmccoltype@colon` {⟨*col type*⟩}

Adds a left or right vertical bar.

```
\newcommand*{\LWR@printmccoltype@colon}[1]{%
    \ifbool{LWR@mcolvertbaronleft}%
        {\defaddtocounter{LWR@mcolvertbarsldash}{1}}% left edge
        {\defaddtocounter{LWR@mcolvertbarsrdash}{1}}% not left edge
}
```

`\LWR@printmccoltype@semicolon` {⟨*col type*⟩}

Adds a left or right vertical bar.

```
\let\LWR@printmccoltype@semicolon\LWR@printmccoltype@colon
```

`\LWR@printmccoltype` {⟨*colspec*⟩} Print any valid column type found. Does not print `@`, `!`, `>`, or `<` columns or their associated tokens.

This is printed as part of the table data tag's `class`.

`\LWR@columntype@mctype@<type>` is defined by `\LWR@modifycolumntype`.

```
\newcommand*{\LWR@printmccoltype}[1]{%
    \LWR@traceinfo{lwr@printmccoltype -#1-}%
```

Get one token of the column spec:

```
    \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
```

Detokenize to avoid problems with special characters:

```
    \edef\LWR@strresult{\detokenize\expandafter{\LWR@strresult}}%
```

Add to the HTML tag depending on which column type is found:

```
    \ifcsdef{LWR@columntype@mctype@\LWR@strresult}%
        {\csuse{LWR@columntype@mctype@\LWR@strresult}}%
        {\boolfalse{LWR@mcolvertbaronleft}}%
    \LWR@traceinfo{lwr@printmccoltype done}%
}
```

`\LWR@printmccoldata@other` {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

For `@`, `!`, `>`, `<`, print the next token without paragraph tags:

```
\newcommand*{\LWR@printmccoldata@other}[2]{%
    \defaddtocounter{LWR@tablemulticolspos}{1}%
    \StrChar{#2}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
    \LWR@strresult%
```

A valid column data type was found:

```
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@printmccoldata@skip` {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

Nothing to print for this column type.

```
\newcommand*{\LWR@printmccoldata@skip}[2]{%
    \defaddtocounter{LWR@tablemulticolspos}{#1}%
```

A valid column data type was found:

```
    \booltrue{LWR@validtablecol}%
}
```

For `\LWR@printmccoldata@...>`, {⟨*num args to skip*⟩} is provided by `\LWR@columntype@mcdata@<coltype>` when it was defined by `\LWR@modifycolumntype`. `\entire colspec` is provided by `\LWR@printmccoldata` when it uses `\LWR@columntype@mcdata@<coltype>`.

`\LWR@printmccoldata@normal` {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

```
\newcommand*{\LWR@printmccoldata@normal}[2]{%
    \LWR@multicoltext%
    \defaddtocounter{LWR@tablemulticolspos}{#1}%
}
```

`\LWR@printmccoldata@paragraph` {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

```
\newcommand*{\LWR@printmccoldata@paragraph}[2]{%
    \LWR@startpars%
    \LWR@multicoltext%
    \defaddtocounter{LWR@tablemulticolspos}{#1}%
    \LWR@stoppars%
}
```

`\LWR@printmccoldata` {⟨*entire colspec*⟩}

Print the data for any valid column type found.

```
\newcommand*{\LWR@printmccoldata}[1]{%
    \LWR@traceinfo{lwr@printmccoldata -#1}%
```

Not yet found a valid column type:

```
    \boolfalse{LWR@validtablecol}%
```

Get one token of the column spec, into a local copy in case nested.

```
    \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
    \edef\LWR@printmccoldatatoken{\LWR@strresult}%
```

Print the text depending on which column type is found. Also handles `@`, `>`, `<` as it comes to them.

```
    \ifcsdef{LWR@columntype@mcdata@\LWR@printmccoldatatoken}%
        {\csuse{LWR@columntype@mcdata@\LWR@printmccoldatatoken}{#1}}%
        {}%
```

If an unknown column type, print the text:

```
    \ifbool{LWR@validtablecol}{}{\LWR@multicoltext{}}%
```

Tracing:

```
    \LWR@traceinfo{lwr@printmccoldata done}%
}
```

\parsemulticolumnalignment {⟨*1: colspec*⟩} {⟨*2: printresults csname*⟩}

Scan the multicolumn specification and execute the printfunction for each entry.

Note that the spec for a p{spec} column, or @, >, <, is a token list which will NOT match l, c, r, or p.

```
\newcommand*{\LWR@parsemulticolumnalignment}[2]{%
    \defcounter{LWR@tablemulticolspos}{1}%
    \StrLen{#1}[\LWR@strresult]%
    \defcounter{LWR@tablemulticolswidth}{\LWR@strresult}%
```

Scan across the tokens in the column spec:

```
    \whileboolexpr{%
        not test {%
            \ifnumcomp{\value{LWR@tablemulticolspos}}{>}%
                {\value{LWR@tablemulticolswidth}}%
        }%
    }%
    {%
```

Execute the assigned print function for each token in the column spec:

```
        \csuse{#2}{#1}%
```

Move to the next token in the column spec:

```
        \defaddtocounter{LWR@tablemulticolspos}{1}%
    }%
}
```

75.22.2 Multicolumn factored code

\LWR@addmulticolvertrulecolor

```
\newcommand*{\LWR@addmulticolvertrulecolor}{%
```

No vertical rules if finishing the tabular with a row of empty cells:

```
    \ifbool{LWR@tabularmutemods}{}{%
```

Left side:

```
        \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{%
            \LWR@tdaddstyle%
            border-left: 1px solid \LWR@vertruleHTMLcolor%
```

```
        }{}%
        \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{%
            \LWR@tdaddstyle%
            border-left: 4px double \LWR@vertruleHTMLcolor%
        }{}%
        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{%
            \LWR@tdaddstyle%
            border-left: 1px dashed \LWR@vertruleHTMLcolor%
        }{}%
        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}{%
            \LWR@tdaddstyle%
            border-left: 2px dashed \LWR@vertruleHTMLcolor%
        }{}%
```

Right side:

```
        \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{%
            \LWR@tdaddstyle%
            border-right: 1px solid \LWR@vertruleHTMLcolor%
        }{}%
        \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{%
            \LWR@tdaddstyle%
            border-right: 4px double \LWR@vertruleHTMLcolor%
        }{}%
        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{%
            \LWR@tdaddstyle%
            border-right: 1px dashed \LWR@vertruleHTMLcolor%
        }{}%
        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}{%
            \LWR@tdaddstyle%
            border-right: 2px dashed \LWR@vertruleHTMLcolor%
        }{}%
    }%
}
```

```
\newcommand{\LWR@multicoltext}{}
```

To find multicolumn right trim:

```
\newcounter{LWR@lastmulticolumn}
```

`\LWR@domulticolumn` `[`⟨*1: vpos*⟩`]` `[`⟨*2: #rows*⟩`]` {⟨*3: numLaTeXcols*⟩} {⟨*4: numHTMLcols*⟩} {⟨*5: colspec*⟩} {⟨*6: text*⟩}

```
\NewDocumentCommand{\LWR@domulticolumn}{o o m m m +m}{%
    \LWR@traceinfo{LWR@domulticolumn -#1- -#2- -#4- -#5-}%
```

Remember the text to be inserted, and when used remember that a valid column type was found:

```
    \renewcommand{\LWR@multicoltext}{%
        #6%
        \booltrue{LWR@validtablecol}%
    }%
```

Expand the preamble and save it.

```
    \LWR@expandpreamble{#5}%
    \edef\LWR@origmccolspec{\the\@temptokena}%
```

Compute the rightmost column to be included. This is used to create the right trim.

```
    \defcounter{LWR@lastmulticolumn}{\value{LWR@tableLaTeXcolindex}}%
    \defaddtocounter{LWR@lastmulticolumn}{#3}%
    \defaddtocounter{LWR@lastmulticolumn}{-1}%
```

Row processing:

```
    \LWR@maybenewtablerow%
```

Begin the opening table data tag:

```
    \LWR@htmltag{%
        td colspan=\textquotedbl#4\textquotedbl\ %

        \IfValueT{#2}{ % rows?
            rowspan=\textquotedbl#2\textquotedbl\ %
        }%

        class=\textquotedbl{}td%
```

Print the column type and vertical bars:

```
        \defcounter{LWR@mcolvertbarsl}{0}%
        \defcounter{LWR@mcolvertbarsr}{0}%
        \defcounter{LWR@mcolvertbarsldash}{0}%
        \defcounter{LWR@mcolvertbarsrdash}{0}%
        \booltrue{LWR@mcolvertbaronleft}%
     \LWR@parsemulticolumnalignment{\LWR@origmccolspec}{LWR@printmccoltype}%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag.

If this position had a "Y" then add "rule" for a horizontal rule:

```
        \LWR@subaddcmidruletrim%
            {%
                \LWR@getexparray{LWR@trimlrules}%
                    {\arabic{LWR@tableLaTeXcolindex}}%
            }%
            {%
                \LWR@getexparray{LWR@trimrrules}%
                    {\arabic{LWR@lastmulticolumn}}%
            }%
```

Also add vertical bar class.

```
        \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{ tvertbarl}{}%
        \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{ tvertbarldouble}{}%
        \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{ tvertbarr}{}%
        \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{ tvertbarrdouble}{}%
        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{ tvertbarldash}{}%
        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}%
            { tvertbarldoubledash}{}%
```

```
        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{ tvertbarrdash}{}%
        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}%
            { tvertbarrdoubledash}{}%
```

Close the class tag's opening quote:

```
        \textquotedbl{}%
```

```
        \LWR@tdstartstyles%
```

Style for vertical position:

```
        \IfValueT{#1}{% vpos?
            \ifstrequal{#1}{b}%
                {%
                    \LWR@tdaddstyle%
                    \LWR@print@mbox{vertical-align:bottom}%
                }{}%
            \ifstrequal{#1}{t}%
                {%
                    \LWR@tdaddstyle%
                    \LWR@print@mbox{vertical-align:top}%
                }{}%
        }% vpos?
```

Style for row colors:

```
        \LWR@addtabularrowcolor%
```

Other styles:

```
        \LWR@addcmidrulewidth%
        \LWR@addcdashline%
        \LWR@addtabularhrulecolor%
        \LWR@addmulticolvertrulecolor%
        \LWR@addformatwpalignment{\LWR@origmccolspec}%
        \LWR@tdendstyles%
    }% end of the opening table data tag
    \boolfalse{LWR@intabularmetadata}%
    \LWR@parsemulticolumnalignment{\LWR@origmccolspec}{LWR@printmccoldata}%
}
```

75.22.3 Multicolumn

`\LWR@htmlmulticolumn` {⟨*numcols*⟩} {⟨*alignment*⟩} {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@htmlmulticolumn}{m m +m}%
{%
```

Figure out how many extra HTML columns to add for `@` and `!` columns:

```
    \LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}%
```

Create the multicolumn tag:

```
    \LWR@domulticolumn{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#3}%
```

Move to the next LaTeX column:

```
    \defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
    \defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing `@` or `!` columns for this cell:

```
    \booltrue{LWR@skipatbang}%
}
```

75.22.4 Longtable captions

longtable captions use `\multicolumn`.

Per the caption package. User-redefinable float type.

```
\providecommand*{\LTcaptype}{table}
```

`\LWR@longtabledatacaptiontag` * [⟨*toc entry*⟩] {⟨*caption*⟩}

```
\NewDocumentCommand{\LWR@longtabledatacaptiontag}{s o +m}
{%
```

Remember the latest name for `\nameref`:

```
    \IfValueTF{#2}{% optional given?
        \ifblank{#2}% optional empty?
        {\LWR@setlatestname{#3}}% empty
        {\LWR@setlatestname{#2}}% given and non-empty
    }% optional given
    {\LWR@setlatestname{#3}}% no optional
```

Create a multicolumn across all the columns:

Figure out how many extra HTML columns to add for `@` and `!` columns found between the first and the last column:

```
    \LWR@tabularhtmlcolumns{1}{\arabic{LWR@tabletotalLaTeXcols}}%
```

Create the multicolumn tag. The caption will be centered by the css caption class.

```
    \LWR@domulticolumn{\arabic{LWR@tabletotalLaTeXcols}}%
        {\arabic{LWR@tabhtmlcoltotal}}%
        {p}%
    {% \LWR@domulticolumn
    \IfBooleanTF{#1}% star?
```

Star version, show a caption but do not make a LOT entry:

```
    {% yes star
        \LWR@figcaption%
        \LWR@isolate{#3}%
        \endLWR@figcaption%
    }%
    {% No star:
```

Not the star version:

Don't step the counter if `\caption[]{A caption.}`

```
\ifbool{LWR@starredlongtable}%
{%
    \ifblank{#2}% TOC entry
    {}%
    {%
        \refstepcounter{\LTcaptype}%
        \protected@edef\@currentlabel{%
            \@nameuse{p@\LTcaptype}\@nameuse{the\LTcaptype}%
        }%
    }%
}{}%
```

Create an HTML caption. Afterwards, maybe make a LOT entry.

```
\LWR@figcaption%
\LWR@isolate{\@nameuse{fnum@\LTcaptype}}%
\CaptionSeparator%
\LWR@isolate{#3}%
\endLWR@figcaption%
```

See if an optional caption was given:

```
\ifblank{#2}% TOC entry empty
```

if the optional caption was given, but empty, do not form a TOC entry

```
{}%
```

If the optional caption was given, but might only be `[]`:

```
{% TOC entry not empty
    \IfNoValueTF{#2}% No TOC entry?
```

The optional caption is `[]`:

```
    {% No TOC entry
        \addcontentsline%
        {\@nameuse{ext@\LTcaptype}}%
        {\LTcaptype}%
        {%
            \protect\numberline%
         {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
            {\ignorespaces \LWR@isolate{#3}\protect\relax}%
        }%
    }% end of No TOC entry
```

The optional caption has text enclosed:

```
    {% yes TOC entry
        \addcontentsline%
        {\@nameuse{ext@\LTcaptype}}%
        {\LTcaptype}%
        {%
            \protect\numberline%
         {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
```

```
                    {\ignorespaces \LWR@isolate{#2}\protect\relax}%
                }%
            }% end of yes TOC entry
        }% end of TOC entry not empty
    }% end of no star
```

Skip any trailing `@` or `!` columns for this cell:

```
    \booltrue{LWR@skipatbang}%
    }% end of \LWR@domulticolumn
  \defaddtocounter{LWR@tableLaTeXcolindex}{\value{LWR@tabletotalLaTeXcols}}%
    \defaddtocounter{LWR@tableLaTeXcolindex}{-1}

}
```

75.22.5 Counting HTML tabular columns

The LaTeX specification for a table includes a number of columns separated by the `&` character. These columns differ in content from line to line. Additional virtual columns may be specified by the special `@` and `!` columns. These columns are identical from line to line, but may be skipped during a multicolumn cell.

For HTML output, `@` and `!` columns are placed into their own tabular columns. Thus, a LaTeX `\multicolumn` command may span several additional `@` and `!` columns in HTML output. These additional columns must be added to the total number of columns spanned by an HTML multi-column data cell.

```
\newcounter{LWR@tabhtmlcolindex}
\newcounter{LWR@tabhtmlcolend}
\newcounter{LWR@tabhtmlcoltotal}
```

`\LWR@subtabularhtmlcolumns {`⟨*index*⟩`}`

Factored from `\LWr@tabularhtmlcolumns`, which follows.

```
\newcommand*{\LWR@subtabularhtmlcolumns}[1]{%
```

Temporarily define a macro equal to the `@` specification for this column:

```
    \edef\LWR@atbangspec{\LWR@getexparray{LWR@colatspec}{#1}}%
```

If the `@` specification is not empty, add to the count:

```
    \ifdefempty{\LWR@atbangspec}%
        {}%
        {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
```

Likewise for the `!` columns:

```
    \edef\LWR@atbangspec{\LWR@getexparray{LWR@colbangspec}{#1}}%
    \ifdefempty{\LWR@atbangspec}%
        {}%
        {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
}
```

`\LWR@tabularhtmlcolumns` {⟨*starting LaTeX column*⟩} {⟨*number LaTeX columns*⟩}

Compute the total number of HTML columns being spanned, considering the starting LaTeX table column and the number of LaTeX tabular columns being spanned. Any `@` and `!` columns within this span are included in the total count. The resulting number of HTML columns is returned in the counter `LWR@tabhtmlcoltotal`.

```
\newcommand*{\LWR@tabularhtmlcolumns}[2]{%
```

Count the starting index, compute ending index, and begin with the count being the LaTeX span, to which additional `@` and `!` columns may be added:

```
    \defcounter{LWR@tabhtmlcolindex}{#1}%
    \defcounter{LWR@tabhtmlcoltotal}{#2}%
    \defcounter{LWR@tabhtmlcolend}{#1}%
    \defaddtocounter{LWR@tabhtmlcolend}{#2}%
```

If at the left edge, add the at/bang columns for the left edge:

```
    \ifnumcomp{\value{LWR@tabhtmlcolindex}}{=}{1}{%
        \LWR@subtabularhtmlcolumns{leftedge}%
    }{}%
```

Walk across the LaTeX columns looking for `@` and `!` columns:

```
    \whileboolexpr{%
        test {%
        \ifnumcomp{\value{LWR@tabhtmlcolindex}}{<}{\value{LWR@tabhtmlcolend}}%
        }%
    }%
    {%
        \LWR@subtabularhtmlcolumns{\arabic{LWR@tabhtmlcolindex}}%
        \defaddtocounter{LWR@tabhtmlcolindex}{1}%
    }% whiledo
}

\end{warpHTML}
```

75.23 Multirow if not loaded

A default defintion in case multirow is not loaded. This is used during table parsing.

```
\begin{warpHTML}
\newcommand{\multirow}[2][c]{}
\end{warpHTML}
```

75.24 Multicolumnrow

A print-mode version is defined here, and is also used during HTML output while inside a `lateximage`.

See section 429 for the HTML versions.

for HTML & PRINT:

```
\begin{warpall}
```

\multicolumnrow {⟨*1:cols*⟩} {⟨*2:halign*⟩} [⟨*3:vpos*⟩] {⟨*4:numrows*⟩} [⟨*5:bigstruts*⟩] {⟨*6:width*⟩} [⟨*7:fixup*⟩] {⟨*8:text*⟩}

For discussion of the use of \DeclareExpandableDocumentCommand, see:
https://tex.stackexchange.com/questions/168434/problem-with-abbreviation-of-multirow-and-multicolumn-latex

\AtBeginDocument to adjust after the user may have loaded multirow, which requires several tests to determine which version is loaded and thus which options are available.

```
\AtBeginDocument{
```

\@ifundefined{@xmultirow} determines if multirow was never loaded.

Null action if not loaded:

```
\@ifundefined{@xmultirow}
{
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {}%
}% no version of multirow was loaded
{% \@xmultirow defined, so some version of multirow was loaded
```

\IfPackageLoadedTF{multirow} determines if v2.0 or later of multirow was used, which included the \ProvidesPackage macro.

The print version:

```
\IfPackageLoadedTF{multirow}{% v2.0 or newer
\IfPackageAtLeastTF{multirow}{2016/09/01}% 2016/09/27 for v2.0
{% v2.0+:
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {\multicolumn{#1}{#2}{\@xmultirow[#3]{#4}[#5]{#6}[#7]{#8}}}%
}
{% loaded but older, probably not executed:
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
}
}% packageloaded{multirow}
```

If not \IfPackageLoadedTF{multirow} but \@xmultirow is defined, then this must be v1.6 or earlier, which did not \ProvidesPackage{multirow}, and did not have the vposn option.

```
{% v1.6 or older did not \ProvidePackage
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
}

}% \@ifundefined{@xmultirow}

\providecommand*{\multicolumnrow}{\LWR@print@multicolumnrow}

}% AtBeginDocument
```

```
\end{warpall}
```

75.25 Utility macros inside a table

for HTML output:

```
\begin{warpHTML}
```

Used to prevent opening a tabular data cell if the following token is one which does not create tabular data:

```
\newcommand*{\LWR@donothing}{}
```

In case array is not loaded:

```
\let\firsthline\relax
\let\lasthline\relax
\newcommand*{\firsthline}{}
\newcommand*{\lasthline}{}
```

In case bigdelim is not loaded:

```
\newcommand*{\ldelim}{}
\newcommand*{\rdelim}{}
```

```
\end{warpHTML}
```

75.26 Special-case tabular markers

for HTML & PRINT:

```
\begin{warpall}
```

`\TabularMacro` Place this just before inserting a custom macro in a table data cell. Doing so tells lwarp not to automatcally start a new HTML table data cell yet. See section 8.10.1.

```
\newcommand*{\TabularMacro}{}
```

```
\end{warpall}
```

`\ResumeTabular` Used to resume tabular entries after resuming an environment.

⚠ `tabular` inside another environment

When creating a new environment which contains a `tabular` environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use `\ResumeTabular` as follows. This is ignored in print mode.

```
\StartDefiningTabulars %   (& is used in a definition)
\newenvironment{outerenvironment}
{
  \tabular{cc}
  left & right \\
}
{
  \TabularMacro\ResumeTabular
  left & right \\
  \endtabular
}
\StopDefiningTabulars
```

for HTML output:

```
\begin{warpHTML}

\newcommand*{\ResumeTabular}{%
    \boolfalse{LWR@exitingtabular}%
    \boolfalse{LWR@tabularmutemods}%
    \boolfalse{LWR@tabularfinalrow}%
    \LWR@getmynexttoken%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}

\newcommand*{\ResumeTabular}{}

\end{warpprint}
```

75.27 Checking for a new table cell

for HTML output:

```
\begin{warpHTML}
```

`\LWR@tabledatacolumntag` Open a new HTML table cell unless the next token is for a macro which does not create data, such as `\hline`, `\toprule`, etc:

```
\newcommand*{\LWR@tabledatacolumntag}%
{%
    \LWR@traceinfo{LWR@tabledatacolumntag}%
```

`\show\LWR@mynexttoken` to see what tokens to look for

If not any of the below, start a new table cell:

```
    \global\let\LWR@mynextaction\LWR@tabledatasinglecolumntag%
```

If find `\end`, exit the tabular:

```
    \ifdefequal{\LWR@mynexttoken}{\end}%
        {%
            \booltrue{LWR@tabularfinalrow}%
            \booltrue{LWR@exitingtabular}%
        }{}%
```

longtable can have a caption in a cell

```
    \ifdefequal{\LWR@mynexttoken}{\caption}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Look for other things which would not start a table cell:

```
    \ifdefequal{\LWR@mynexttoken}{\multicolumn}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
    \ifdefequal{\LWR@mynexttoken}{\multirow}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
    \ifdefequal{\LWR@mynexttoken}{\multicolumnrow}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
    \ifdefequal{\LWR@mynexttoken}{\noalign}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mrowcell, this is a cell to be skipped over:

```
    \ifdefequal{\LWR@mynexttoken}{\mrowcell}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mcolrowcell, this is a cell to be skipped over:

```
    \ifdefequal{\LWR@mynexttoken}{\mcolrowcell}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\TabularMacro}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\hline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\firsthline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\lasthline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\toprule}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\midrule}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\cmidrule}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\morecmidrules}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\specialrule}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\cline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

```
    \ifdefequal{\LWR@mynexttoken}{\bottomrule}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\hhline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\rowcolor}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\arrayrulecolor}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\doublerulesepcolor}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\warpprintonly}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\warpHTMLonly}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\ldelim}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\rdelim}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

For arydshln:

```
    \ifdefequal{\LWR@mynexttoken}{\hdashline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\cdashline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\firsthdashline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%

    \ifdefequal{\LWR@mynexttoken}{\lasthdashline}%
        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Ignore an empty line between rows:

```
    \ifdefequal{\LWR@mynexttoken}{\par}%
        {%
            \global\let\LWR@mynextaction\LWR@donothing%
        }{}%
```

No action for an \end token.

Add similar to the above for any other non-data tokens which might appear in the table.

Start the new table cell if was not any of the above:

```
   \LWR@traceinfo{LWR@tabledatacolumntag: done, about to do LWR@mynextaction}%
    \LWR@mynextaction%
}
```

```
\end{warpHTML}
```

75.28 \mrowcell

for HTML & PRINT!

```
\begin{warpall}
```

`\mrowcell` The user must insert `\mrowcell` into any `\multirow` cells which must be skipped.
⚠ multirow cells This command has no action during print output.

```
\newcommand*{\mrowcell}{}
```

```
\end{warpall}
```

75.29 \mcolrowcell

for HTML & PRINT!

```
\begin{warpall}
```

`\mcolrowcell` The user must insert `\mcolrowcell` into any `\multicolumnrow` cells which must
⚠ multirow cells be skipped. This command has no action during print output.

```
\newcommand*{\mcolrowcell}{}
```

```
\end{warpall}
```

75.30 HTML tabular environment

for HTML output!

```
\begin{warpHTML}
```

These are default defininitions in case booktabs is not loaded, and are not expected to used, but must exist as placeholders. memoir may have already loaded booktabs.

```
\providecommand*{\toprule}[1][]{\hline}
\providecommand*{\midrule}[1][]{\hline}
\providecommand*{\cmidrule}{\cline}
\providecommand*{\bottomrule}[1][]{\hline}
\providecommand*{\addlinespace}[1][]{}
\providecommand*{\morecmidrules}{}
\providecommand*{\specialrule}[3]{\hline}
```

`\noalign` {⟨*text*⟩} Redefined for use inside `tabular`.

```
\LetLtxMacro\LWR@orignoalign\noalign

\newcommand{\LWR@tabularnoalign}[1]{%
    \advance\rownum\m@ne%
    \LetLtxMacro\LWR@save@xcolorrowHTMLcolor\LWR@xcolorrowHTMLcolor%
    \renewcommand*{\LWR@xcolorrowHTMLcolor}{}%
    \multicolumn{\value{LWR@tabletotalLaTeXcols}}{l}{#1} \\
    \LetLtxMacro\LWR@xcolorrowHTMLcolor\LWR@save@xcolorrowHTMLcolor%
    % \@rowc@lors%
    \LWR@getmynexttoken%
}
```

`\LWR@HTMLhline` The definition of `\hline` depends on whether tabls has been loaded. If so, optional space below the line may be specified, but will be ignored.

```
\AtBeginDocument{

\IfPackageLoadedTF{lwarp-tabls}
{
    \newcommand*{\LWR@HTMLhline}[1][]{%
        \ifbool{FormatWP}%
            {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{LWR@hlines}{1}}%
        \LWR@getmynexttoken}%
}
{
    \newcommand*{\LWR@HTMLhline}{%
        \ifbool{FormatWP}%
            {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{LWR@hlines}{1}}%
        \LWR@getmynexttoken}%
}

}% AtBeginDocument
```

`\LWR@HTMLcline {`⟨*columns*⟩`}`

```
\NewDocumentCommand{\LWR@HTMLcline}{m}%
{%
    \LWR@docmidrule{#1}%
    \LWR@maybenewtablerow%
    \LWR@getmynexttoken%
}%
```

`\LWR@tabular@warpprintonly {`⟨*contents*⟩`}`

Only process the contents if producing printed output. Modified inside a `tabular` to grab the next token.

```
\newcommand{\LWR@tabular@warpprintonly}[1]{%
    \ifbool{warpingprint}{#1}{}%
    \LWR@getmynexttoken%
}
```

`\LWR@nullifyNoAutoSpacing` For babel-french, turn off auto spacing at the start of the tabular, then nullify the autospacing commands inside the tabular, since they were not compatible with the tabular parsing code for each cell, which uses xstring.

```
\AtBeginDocument{
\@ifundefined{NoAutoSpacing}%
{% no babel-french
    \newcommand*{\LWR@nullifyNoAutoSpacing}{}
}% no babel-french
{% yes babel-french
    \newcommand*{\LWR@nullifyNoAutoSpacing}{%
        \NoAutoSpacing%
        \renewcommand*{\NoAutoSpacing}{}%
        \renewcommand*{\LWR@FBcancel}{}%
    }
```

```
}% yes babel-french
}% AtBeginDocument
```

`tabular` (*env.*) `<direction>` [⟨*vertposition*⟩] {⟨*colspecs*⟩}

The `<direction>` is from plext for Japanese documents, and is ignored.

```
\StartDefiningTabulars

\NewDocumentCommand{\LWR@HTML@@tabular}{d<> o m}
{%
    \LWR@traceinfo{LWR@HTML@@tabular started}%
```

⚠ `<table>` inside `<span>` In LaTeX, a `tabular` may be placed inside a `minipage`, but in HTML a `<table>` may not be inside a `<span>`. Since there may be several nested `<span>`s, with an unknown number of other objects between, it is hard to undo all these `<span>`s before the `<table>` then redo them after. The broswer probably compensates for this situation, but formatting may be lost inside the `<table>` because several things are neutralized inside a `<span>`. Furthermore, in the HTML output, the entire `<table>` is placed on a single line of HTML code, since the line breaking commands are neutralized inside a `<span>`. Since this is such a sloppy situation, a warning is issued here instructing the user to please isolate the `<span>` to print-only.

```
    \LWR@spanwarnformat{tabular}%
    \addtocounter{LWR@tabulardepth}{1}%
```

Not yet started a table row:

```
    \boolfalse{LWR@startedrow}%
```

Not yet doing any rules:

```
    \defcounter{LWR@hlines}{0}%
    \defcounter{LWR@hdashedlines}{0}%
    \boolfalse{LWR@doingtbrule}%
    \boolfalse{LWR@doingcmidrule}%
```

For babel-french, turn off auto spacing one time, then nullify the autospacing commands since were not compatible with the tabular parsing code.

```
    \LWR@nullifyNoAutoSpacing%
```

Have not yet found the end of tabular command. Unmute the `@` and `!` columns.

```
    \boolfalse{LWR@exitingtabular}%
    \boolfalse{LWR@tabularmutemods}%
```

Not adding final row for the lower border:

```
    \boolfalse{LWR@tabularfinalrow}%
```

Error if failed to use `\mrowcell` or `\mcolrowcell` when needed.

```
    \boolfalse{LWR@usedmultirow}%
    \boolfalse{LWR@foundmrowcell}%
```

In case of nesting:

```
\renewcommand*{\LWR@multicoltext}{}%
\booltrue{LWR@intabularmetadata}%
```

New PDF page, unless in a `\multirow`:

```
\ifbool{LWR@in@multirow@par}%
    {\leavevmode\LWR@orignewline}%
    {\LWR@forcenewpage}%
```

In case of nesting, locally no longer in a `\multirow`:

```
\boolfalse{LWR@in@multirow@par}%
```

Create the `table` tag:

```
\LWR@htmlblocktag{table}%
```

Parse the table columns:

```
\LWR@parsetablecols{#3}%
```

Table col spec is: `\LWR@tablecolspec` which is a string of `llccrr`, etc.

Do not place the table inside a paragraph:

```
\LWR@stoppars%
```

Without at least one header cell, some screen readers think that the table is just for page layout, and do not read it as data. Add a hidden row with a single non-empty header cell to tell the screen readers that this is a table of data for the user.

```
\LWR@htmltag{tr style="display:none"}%
\LWR@htmltag{th}.\LWR@htmltag{/th}%
\LWR@htmltag{/tr}%
\LWR@orignewline%
\LWR@forceemptyline%
```

Track column #:

```
\defcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
\global\boolfalse{LWR@tabularcelladded}%
```

Start looking for midrules:

```
\LWR@clearmidrules%
```

`\\` becomes a macro to end the table row:

```
\LetLtxMacro{\\}{\LWR@tabularendofline}%
```

`\warpprintonly` inside a `tabular` must grab the next token.

```
\LetLtxMacro\warpprintonly\LWR@tabular@warpprintonly%
```

The following adjust for colortbl.

```
    \LetLtxMacro\arrayrulecolor\arrayrulecolornexttoken%
    \LetLtxMacro\doublerulesepcolor\doublerulesepcolornexttoken%
    \def\LWR@columnHTMLcolor{}%
    \def\LWR@rowHTMLcolor{}%
    \def\LWR@cellHTMLcolor{}%
    \@rowcolors%
```

The vertical rules are set to the color active at the start of the tabular. `\arrayrulecolor` will then affect horizontal rules inside the tabular, but not the vertical rules.

```
    \ifdefvoid{\LWR@ruleHTMLcolor}%
        {\edef\LWR@vertruleHTMLcolor{black}}%
        {\edef\LWR@vertruleHTMLcolor{\LWR@origpound\LWR@ruleHTMLcolor}}%
```

Tracking the depth of cell color s:

```
    \defcounter{LWR@cellcolordepth}{0}%
```

The following may appear before a data cell is created, so after doing their actions, we look ahead with `\LWR@getmynextoken` to see if the next token might create a new data cell:

The optional parameter for `\hline` supports the tabls package.

```
    \LWR@traceinfo{LWR@@HTML@tabular: redefining macros}%
    \LetLtxMacro\noalign\LWR@tabularnoalign%
    \LetLtxMacro\hline\LWR@HTMLhline%
    \LetLtxMacro\cline\LWR@HTMLcline%

    \DeclareDocumentCommand{\hdashline}{o}{%
        \ifbool{FormatWP}%
            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{LWR@hdashedlines}{1}}%
        \LWR@getmynexttoken%
    }%

    \DeclareDocumentCommand{\cdashline}{m}{%
        \LWR@docdashline{##1}\LWR@getmynexttoken%
    }%

    \DeclareDocumentCommand{\firsthdashline}{o}{%
        \ifbool{FormatWP}%
            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{LWR@hdashedlines}{1}}%
        \LWR@getmynexttoken%
    }%

    \DeclareDocumentCommand{\lasthdashline}{o}{%
        \ifbool{FormatWP}%
            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{LWR@hdashedlines}{1}}%
        \LWR@getmynexttoken%
    }%
```

The following create data cells and will have no more data in this cell, so we do not want to look ahead for a possible data cell, so do not want to use `\LWR@getmynexttoken`.

```
    \renewcommand{\multicolumn}{\LWR@htmlmulticolumn}%
    \renewcommand*{\mrowcell}{%
        \LWR@maybenewtablerow%
        \LWR@tabularleftedge%
        \booltrue{LWR@skippingmrowcell}%
        \booltrue{LWR@foundmrowcell}%
    }%
    \renewcommand*{\mcolrowcell}{%
        \LWR@maybenewtablerow%
        \booltrue{LWR@skippingmcolrowcell}%
        \booltrue{LWR@foundmrowcell}%
    }%
    \LetLtxMacro\caption\LWR@longtabledatacaptiontag%
```

Reset for new processing:

```
    \boolfalse{LWR@tableparcell}%
    \boolfalse{LWR@skippingmrowcell}%
    \boolfalse{LWR@skippingmcolrowcell}%
    \boolfalse{LWR@skipatbang}%
    \boolfalse{LWR@emptyatbang}%
```

Set `&` for its special meaning inside the tabular:

```
    \StartDefiningTabulars%
    \protected\gdef&{\LWR@tabularampersand}%
```

Locally force any minipages to be fullwidth, until the end of the `tabular`:

```
    \booltrue{LWR@forceminipagefullwidth}%
```

Nest one level deeper of tabular paragraph handling:

```
    \addtocounter{LWR@tabularpardepth}{1}%
```

Look ahead for a possible table data cell:

```
    \LWR@traceinfo{LWR@@HTML@tabular: about to LWR@getmynexttoken}%
    \LWR@getmynexttoken%
}%
```

Ending the environment:

```
\newcommand*{\LWR@HTML@endtabular}
{%
    \LWR@traceinfo{LWR@HTML@endtabular}%
```

Unnest one level of tabular paragraph handling:

```
    \addtocounter{LWR@tabularpardepth}{-1}%
```

Finish a row which is not yet done:

```
    \ifboolexpr{%
```

```
        test {%
            \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}%
                {\value{LWR@tabletotalLaTeXcols}}
        } or %
        (%
            bool{LWR@intabularmetadata} and%
            not bool{LWR@tabularcelladded} and%
            test {%
                \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}%
                    {\value{LWR@tabletotalLaTeXcols}}%
            }%
        )%
    }%
    {%

        \booltrue{LWR@tabularfinalrow}%
        \LWR@tabularfinishrow%
        \boolfalse{LWR@tabularfinalrow}%
    }%
    {%
        \LWR@closetabledatacell%
    }%
    \ifbool{LWR@startedrow}%
        {\LWR@htmltag{/tr}\LWR@orignewline}%
        {}%
```

xcolor row color support:

```
    \@rowc@lors%

    \LWR@htmlblocktag{/table}%
    \boolfalse{LWR@intabularmetadata}%
```

Unnest one level of tabular:

```
    \addtocounter{LWR@tabulardepth}{-1}%
```

Restore & to its usual meaning:

```
    \ifnumequal{\value{LWR@tabulardepth}}{0}{%
        \protected\gdef&{\LWR@origampmacro}%
        \StopDefiningTabulars%
    }{}%
```

Error if used `\multirow` or `\multicolumnrow` without using `\mrowcell` or `\mcolrowcell`.

```
    \ifbool{LWR@usedmultirow}{%
        \ifbool{LWR@foundmrowcell}%
            {\relax}%
            {%
                \PackageError{lwarp}%
                {%
              When using \protect\multirow, \protect\multicolumnrow,\MessageBreak
                     or the bigdelim package,\MessageBreak
              place \protect\mrowcell\space or \protect\mcolrowcell\MessageBreak
                     in empty cells which are to be skipped.\MessageBreak
                     See the Lwarp package documentation:\MessageBreak
```

```
                    "Special cases and limitations" -> "Tabular"
                }%
                {%
                    See the Lwarp package documentation:\MessageBreak
                    "Special cases and limitations" -> "Tabular".
                }%
            }%
    }{}%

    \LWR@traceinfo{LWR@HTML@endtabular finished}%
}

\csletcs{LWR@HTML@endtabular*}{LWR@HTML@endtabular}

\StopDefiningTabulars
```

siunitx may redefine `tabular`, so set the following later:

```
\AtBeginDocument{
    \LetLtxMacro\LWR@origendtabular\endtabular
    \csletcs{LWR@origendtabular*}{endtabular*}
    \LWR@formatted{@tabular}
    \LWR@formatted{endtabular}
    \LWR@formatted{endtabular*}
}

\end{warpHTML}
```

76 Cross-references

Sectioning commands have been emulated from scratch, so the cross-referencing commands are custom-written for them. Emulating both avoids several layers of patches.

`*_html.aux` (*file*) A new entry in `*_html.aux` is used to remember section name, file, and lateximage depth and number for each label:

```
\newlabel{<labelname>@lwarp}{{<section name>}{<filename>}
                                {<limagedepth>}{<limagenumber>}}
```

Table 16 shows the data structures related to cross-referencing.

for HTML output:

```
\begin{warpHTML}
```

76.1 Setup

`\@currentlabelname` To remember the most recently defined section name, description, or caption, for `\nameref`.

```
\def\@currentlabelname{\linkhomename}%
```

`\LWR@stripperiod` {⟨*text*⟩} [⟨.⟩]

Table 16: Cross-referencing data structures

Original LaTeX: (print and HTML)

\refstepcounter: Steps the couunter and sets `\@currentlabel`.

\@currentlabel: `\p@<ctr>\the<ctr>` Updated by `\refstepcounter`.

\label: Writes to the `.aux` file:
`\newlabel{<label>}{{\@currentlabel}{\thepage}}`

\newlabel: When the `.aux` file is read, sets `\r@<label>`.

\r@<label>: Set to: `{{\@currentlabel}{\thepage}}`

\ref: Returns the first part of `\r@<label>`.

\pageref: Returns the second part of `\r@<label>`.

Added by lwarp: (HTML only)

\label: Adds HTML tags (section 76.3), and another `.aux` entry (section 76.2). If memoir is used, its `\@mem@old@label` points to lwarp's version, and cleveref patches.

\newlabel: Unchanged. When the `.aux` file is read, sets `\r@<label>@lwarp`.

\r@<label>@lwarp: Set to `{{section_name}{file_name}{depth}{number}}`:

\LWR@nameref: The section or object name for this label.

\LWR@currentautosecpageref: The `LWR@currentautosecpage` for this label.

\LWR@htmlfileref: The filenumber or name for this label.

\LWR@lateximagedepthref: The `lateximagedepth` for this label.

\LWR@lateximagenumberref: The `lateximagenumber` for this label.

\nameref: Emualted from hyperref for lwarp. See section 76.4.

\ref and \nameref: Adds HTML tags. See section 76.4.

Added by amsmath: (print and HTML)

\label: Execution is delayed until the math environment is completed.

\ltx@label: LaTeX `\label`, (HTML: patched by lwarp,) later patched by cleveref.

Added by cleveref: (print and HTML)

\refstepcounter: Added: sets `\cref@currentlabel`.

\cref@currentlabel: (`<type>=<ctr>` unless an alias is used):
`[<type>][\arabic{<ctr>}][<parent ctrs>]{\p@<ctr>\the<ctr>}` Also see section 60.4 for use with footnotes.

\label: Writes to the `.aux` file:
`\newlabel{<label>@cref}{{\cref@currentlabel}{\thepage}}`

\newlabel: Unchanged. When the `.aux` file is read, sets `\r@<label>@cref`.

\r@<label>@cref: Set to: `{{\cref@currentlabel}{\thepage}}`

Utility functions: See `\cref@getlabel`, `\cref@gettype`, `\cref@getcounter`, `\cref@getprefix`.

Cross-referencing names: `\crefname` and `\Crefname` assign human-readable names for references to this counter type.

Additionally patched by lwarp: (HTML only)

\cref, etc.: Modified for lwarp. See section 202.

\label inside math: See section 83.7.1.

Footnotes: See `\noteentry` in section 60.4.

Removes a trailing period.

```
\def\LWR@stripperiod#1.\ltx@empty#2\@nil{#1}%
```

`\LWR@setlatestname` {⟨*object name*⟩}

Removes `\label`, strips any final period, and remembers the result.

```
\newcommand*{\LWR@setlatestname}[1]{%
```

Remove `\label` and other commands from the name, the strip any final period. See gettitlestring.

```
    \GetTitleStringExpand{#1}%
    \edef\@currentlabelname{\detokenize\expandafter{\GetTitleStringResult}}%
    \edef\@currentlabelname{%
        \expandafter\LWR@stripperiod\@currentlabelname%
        \ltx@empty.\ltx@empty\@nil%
    }%
}
```

76.2 New lwarp labels.

`*_html.aux` (*file*) A new entry in `*_html.aux` is used to remember section name, file, and latexmage depth and number for each label:

```
\newlabel{<labelname>@lwarp}{{<section name>}{<filename>}
                                    {<limagedepth>}{<limagenumber>}}
```

See:
http://tex.stackexchange.com/questions/57194/extract-section-number-from-equation-reference

`\LWR@setref` {⟨*args list*⟩} {⟨*selector*⟩} {⟨*label*⟩}

`\@setref` without the `\null` (`\hbox`), and without the warning messages. Each caused problems with lwarp references. The regular reference will cause the warning.

```
\def\LWR@setref#1#2#3{%
  \ifx#1\relax%
    ??%
  \else%
   \expandafter#2#1%
  \fi}
```

`\LWR@nameref` {⟨*label*⟩} Returns the section name for this label:

```
\newcommand*{\LWR@nameref}[1]{%
    \begingroup%
    \LWR@nullifyfootnotes%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@firstoffive{#1}%
    \endgroup%
}
```

`\LWR@currentautosecpageref {⟨label⟩}` Returns the `LWR@currentautosecpage` for this label:

```
\newcommand*{\LWR@currentautosecpageref}[1]{%
   \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@secondoffive{#1}%
}
```

`\LWR@htmlfileref {⟨label⟩}` Returns the file number or name for this label:

```
\newcommand*{\LWR@htmlfileref}[1]{%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@thirdoffive{#1}%
}
```

`\LWR@lateximagedepthref {⟨label⟩}` Returns the `lateximagedepth` for this label:

```
\newcommand*{\LWR@lateximagedepthref}[1]{%
   \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@fourthoffive{#1}%
}
```

`\LWR@lateximagenumberref {⟨label⟩}` Returns the `lateximagenumber` for this label:

```
\newcommand*{\LWR@lateximagenumberref}[1]{%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@fifthoffive{#1}%
}
```

`\LWR@write@lwarplabel {⟨label⟩}` Sanitize the name and then creates the label:

```
\newcommand*{\LWR@write@lwarplabel}[1]{%
    \LWR@traceinfo{LWR@write@lwarplabel !#1!}%
    \LWR@setlatestname{\@currentlabelname}%
        \@bsphack%
        \protected@write\@auxout{}%
            {%
                \string\newlabel{#1@lwarp}{%
                    {\@currentlabelname}%
                    {\theLWR@currentautosecpage}%
                    {%
                        \ifbool{FileSectionNames}%
                            {\LWR@thisfilename}%
                            {\arabic{LWR@htmlfilenumber}}%
                    }%
                    {\arabic{LWR@lateximagedepth}}%
                    {\arabic{LWR@lateximagenumber}}%
                }%
            }%
        \@esphack%
}
```

76.3 Labels

`\LWR@label@subcreatetag` Creates the tag from `\LWR@sanitized`.

```
\newcommand*{\LWR@label@subcreatetag}{%
   \LWR@htmltag{a \LWR@print@mbox{id=\textquotedbl\LWR@sanitized\textquotedbl}}%
    \LWR@htmltag{/a}%
}
```

\LWR@label@inmathcomment

```
\newcommand*{\LWR@label@inmathcomment}{%
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
    {%
```

The combined LaTeX & HTML label is printed in a \mbox field:

```
        \mbox{%
```

Shift the label over to the right side of the environment to avoid over-printing the math:

```
        \ifdef{\totwidth@}{\ifbool{LWR@amsmultline}{}{\hspace*{\totwidth@}}}{}%
```

Temporarily end the HTML comment, insert the LaTeX & HTML label, then resume the HTML comment. \@firstofone is required to remove extra braces introduced by the amsmath package.)

```
            \LWR@htmlclosecomment%
            \LWR@label@subcreatetag%
            \LWR@htmlopencomment%
        }% mbox
    }% mathjax
    {%
        \LWR@label@subcreatetag%
    }%
}
```

\LWR@label@createtag {⟨*label*⟩} Creates an HTML id tag.

Used by \LWR@new@label and \hyperdef.

\detokenize is used to allow underscores in the labels.

```
\newcommand*{\LWR@label@createtag}[1]{%
    \LWR@traceinfo{LWR@label@createtag !#1!}%
```

Create an HTML id tag unless are inside a lateximage, since it would appear in the image:

```
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
    {}%
    {% not lateximage
```

If not doing a lateximage, create an HTML ID tag.

```
        \LWR@sanitize{#1}%
        \ifbool{LWR@insidemathcomment}%
        {% inside HTML math comment
            \LWR@label@inmathcomment%
        }% inside HTML math comment
        {% not inside HTML math comment
            \ifbool{LWR@doingstartpars}%
            {% pars allowed
                \ifbool{LWR@doingapar}%
                {% par started
                    \LWR@label@subcreatetag%
```

```
                }% par started
                {% par not started
                    \LWR@stoppars%
                    \LWR@label@subcreatetag%
                    \LWR@startpars%
                }% par not started
            }% pars allowed
            {% pars not allowed
                    \LWR@label@subcreatetag%
            }% pars not allowed
        }% not inside HTML math comment
    }% not lateximage
}
```

`\LWR@new@label {⟨label⟩}`

`\label` during HTML output when not in SVG math mode, removing extra spaces around the label, as done by a regular LaTeX `\label`.

The is also used during a `lateximage`, including SVG math, since the special label handling is required, but `\LWR@label@createtag` does not generate HTML tags inside a `lateximage`.

If memoir is used, it's `\@mem@old@label` is pointed here.

clevereref later encases this to add its own cross-referencing, and also patches memoir.

```
\newcommand*{\LWR@new@label}[1]{%
    \LWR@traceinfo{LWR@new@label: starting}%
    \LWR@traceinfo{LWR@new@label: !#1!}%
% \@bsphack%
```

Create a traditional LaTeX label, as modified by cleveref:

```
    \LWR@orig@label{#1}%
```

Create a special label which holds the section number, section name, `LWR@htmlfilenumber`, `LWR@lateximagedepth`, and `LWR@lateximagenumber`:

```
    \LWR@traceinfo{%
        LWR@new@label: filesectionnames is %
        \ifbool{FileSectionNames}{true}{false}%
    }%
    \LWR@traceinfo{%
        LWR@new@label: LWR@thisfilename is !\LWR@thisfilename!%
    }%
    \LWR@traceinfo{%
        LWR@new@label: LWR@htmlfilenumber is \arabic{LWR@htmlfilenumber}%
    }%
    \LWR@write@lwarplabel{#1}%
    \LWR@label@createtag{#1}%
    % \@esphack%
    \LWR@traceinfo{LWR@new@label: done}%
}
```

76.4 References

\LWR@addlinktitle

```
\newcommand*{\LWR@addlinktitle}{%
    \ifdefvoid{\LWR@ThisAltText}{}{ % space
        title=\textquotedbl\LWR@ThisAltText\textquotedbl\ % space
        \gdef\LWR@ThisAltText{}%
    }%
}
```

\LWR@startref {⟨*label*⟩} (Common code for \ref and \nameref.)

Open an HTML tag reference to a filename, # character, and a label.

```
\newcommand*{\LWR@startref}[1]
{%
    \LWR@sanitize{#1}%
    \LWR@traceinfo{LWR@startref A: !#1!}%
```

Create the filename part of the link:

```
    \LWR@htmltag{a href=\textquotedbl%
    \LWR@traceinfo{LWR@startref B}%
    \LWR@print@mbox{\LWR@htmlrefsectionfilename{#1}}%
    \LWR@traceinfo{LWR@startref C}%
    \LWR@origpound%
```

Create the destination id:

See if LWR@lateximagedepth is unknown:

```
    \LWR@traceinfo{LWR@startref D: !#1!}%
    \ifcsundef{r@#1@lwarp}%
```

"??" if LWR@lateximagedepth is unknown, so create a link with an unknown destination:

```
    {%
        \LWR@traceinfo{LWR@startref D0: ??}%
        ??%
    }%
```

If LWR@lateximagedepth is known. Use a latexmage if the depth is greater than zero, or a regular link otherwise:

(Using xifthen \ifthenelse here failed in some cases, but etoolbox \ifnumgreater works.)

```
    {%
        \ifnumgreater{\LWR@lateximagedepthref{#1}}{0}%
            {%

                lateximage-\BaseJobname-\LWR@lateximagenumberref{#1}%
            }%
            {%
                \LWR@traceinfo{LWR@startref D3}%
```

\detokenize is used to allow underscores in the labels:

```
                \LWR@print@mbox{\LWR@sanitized}%
            }%
    }%
    \LWR@traceinfo{LWR@startref E}%
```

Closing quote:

```
    \textquotedbl%
```

Maybe add a title:

```
    \LWR@addlinktitle%
    }%
    \LWR@traceinfo{LWR@startref F}%
}
```

\LWR@subnewref {⟨*label*⟩} {⟨*label or sub@label*⟩}

Factored for the subfig package. Uses the original label for the hyper-reference, but prints its own text, such as "1(b)".

```
\NewDocumentCommand{\LWR@subnewref}{m m}{%
    \LWR@traceinfo{LWR@subnewref #1 #2}%
    \LWR@startref{#1}%
    \LWR@print@ref{#2}%
    \LWR@htmltag{/a}%
}
```

\ref * {⟨*label*⟩}

\ref is redefined to \LWR@HTML@ref, except inside the text part of a \hyperref, where it is redefined to \LWR@ref@ignorestar.

\LWR@HTML@ref * {⟨*label*⟩} Create an internal document reference link, or without a link if starred per hyperref.

The HTML version:

```
\NewDocumentCommand{\LWR@HTML@ref}{s m}{%
    \LWR@traceinfo{LWR@HTML@ref !#2!}%
    \IfBooleanTF{#1}%
        {\LWR@print@ref{#2}}%
        {\LWR@subnewref{#2}{#2}}%
}

\LWR@formatted{ref}


\NewDocumentCommand{\LWR@HTML@Ref}{s m}{%
    \LWR@traceinfo{LWR@HTML@Ref !#2!}%
    \IfBooleanTF{#1}%
        {\LWR@print@Ref{#2}}%
        {\LWR@subnewref{#2}{#2}}%
}

\LWR@formatted{Ref}
```

\LWR@refwithsection * {⟨*label*⟩}

Creates a reference, using the section number as the text. Used for back references.

```
\NewDocumentCommand{\LWR@refwithsection}{s m}{%
    \LWR@traceinfo{LWR@refwithsection !#2!}%
```

If starred, just use the text without a hyperlink:

```
    \IfBooleanTF{#1}%
     {\LWR@print@ref{\BaseJobname-autopage-\LWR@currentautosecpageref{#2}}}%
        {%
```

Open the reference:

```
            \LWR@startref{#2}%
```

Add the text of the link.

Check for and handle an undefined reference:

```
            \edef\@tempa{\LWR@currentautosecpageref{#2}}%
            \ifdefstring{\@tempa}{??}%
                {??}%
```

For a defined reference:

```
                {%
```

Set `\@tempa` to `\r@<label>`, which is `{section number}{page number}`.

```
                    \edef\@tempa{%
                        \csname
                     r@\BaseJobname-autopage-\LWR@currentautosecpageref{#2}%
                        \endcsname%
                    }%
```

Check the section number alone:

```
                    \edef\@tempa{\expandafter\@firstoftwo\@tempa}%
```

If the reference has no section number print an asterisk:

```
                    \expandafter\ifblank\expandafter{\@tempa}%
                        {*}
```

If there is a section number, print it:

```
                        {%
                            \LWR@print@ref{%
                       \BaseJobname-autopage-\LWR@currentautosecpageref{#2}%
                            }%
                        }%
            }%
```

Close the reference:

```
            \LWR@htmltag{/a}%
        }%
}
```

For MathJax:

```
\CustomizeMathJax{\let\LWRref\ref}
\CustomizeMathJax{\renewcommand{\ref}{\ifstar\LWRref\LWRref}}
```

`\pagerefPageFor` Text for page references.

```
\newcommand*{\pagerefPageFor}{see }
```

`\pageref * {⟨label⟩}` Create an internal document reference, or just the unlinked number if starred, per hyperref.

```
\NewDocumentCommand{\LWR@new@pageref}{s m}{%
    \IfBooleanTF{#1}%
        {(\pagerefPageFor\LWR@print@ref{#2})}%
        {(\cpageref{#2})}%
}
```

`\nameref {⟨label⟩}`

```
\newrobustcmd*{\nameref}[1]{%
    \LWR@traceinfo{nameref}%
    \LWR@startref{#1}%
    \LWR@traceinfo{nameref B}%
    \LWR@nameref{#1}%
    \LWR@traceinfo{nameref C}%
    \LWR@htmltag{/a}%
    \LWR@traceinfo{nameref: done}%
}
```

`\Nameref {⟨label⟩}` In print, adds the page number. In HTML, does not.

```
\LetLtxMacro\Nameref\nameref
```

76.5 Hyper-references

⚠ Note that the code currently only sanitizes the underscore character. Additional characters should be rendered inert as well. See the `hyperref.sty` definition of `\gdef\hyper@normalise` for an example.

`hyperref` (*Pkg*)

⚠ Do not tell other packages that hyperref is emulated. Some packages patch various commands if hyperref is present, which will probably break something, and the emulation already handles whatever may be emulated anyhow.

```
% DO NOT TELL OTHER PACKAGES TO ASSUME HYPERREF, lest they attempt to patch it:
% \EmulatesPackage{hyperref}[2015/08/01]% Disabled.  Do not do this.
```

Emulates hyperref:

\@currentHref Added to support backref.

```
\AtBeginDocument{
    \def\@currentHref{\BaseJobname-autopage-\theLWR@previousautopagelabel}%
}
```

\LWR@linkcatcodes Sets catcodes before processing macros which have hyperlinks as arguments.

```
\newcommand*{\LWR@linkcatcodes}{%
    \catcode`\#=12%
    \catcode`\%=12%
    \catcode`\&=12%
    \catcode`\~=12%
    \catcode`\_=12%
```

For babel-french:

```
    \LWR@hook@processingtags%
}
```

\LWR@linkmediacatcodes Sets catcodes before processing macros which have hyperlinks as arguments. Modified for multimedia links.

```
\newcommand*{\LWR@linkmediacatcodes}{%
    \catcode`\#=12%
    \catcode`\%=12%
%   \catcode`\&=12% left alone for splitting flash variables
    \catcode`\~=12%
    \catcode`\_=12%
```

For babel-french:

```
    \LWR@hook@processingtags%
}
```

\LWR@subhyperref {⟨*URL*⟩}

Starts a link for \LWR@hrefb. A group must have been opened first, with nullified catcodes. The text name is printed afterwards, after the group is closed and catcodes restored.

```
\NewDocumentCommand{\LWR@subhyperref}{m}{%
    \LWR@traceinfo{LWR@subhyperref !#1!}%
    \LWR@sanitize{#1}%
    \LWR@htmltag{%
        a href=\textquotedbl\LWR@sanitized\textquotedbl\ % space
        \LWR@addlinktitle % space
        target=\textquotedbl\_{}blank\textquotedbl\ % space
    }%
}
```

\LWR@subhyperreftext {⟨*text*⟩}

Finishes the hyperref for \LWR@hrefb. Catcodes must have been restored already. To be used after \LWR@subhyperref, and after its group has been closed.

```
\newcommand{\LWR@subhyperreftext}[1]{%
    #1%
    \LWR@htmltag{/a}%
    \LWR@ensuredoingapar%
}
```

\LWR@subhyperrefclass {⟨*URL*⟩} {⟨*text*⟩} {⟨*htmlclass*⟩}

```
\NewDocumentCommand{\LWR@subhyperrefclass}{m +m m}{%
    \LWR@htmltag{%
        a % space
     href=\textquotedbl\begingroup\@sanitize#1\endgroup\textquotedbl\ % space
        class=\textquotedbl#3\textquotedbl\ % space
        \LWR@addlinktitle % space
    }\LWR@orignewline%
    #2%
    \LWR@htmltag{/a}%
    \LWR@ensuredoingapar%
}
```

\LWR@href [⟨*options*⟩] {⟨*URL*⟩} {⟨*text*⟩}

Create a link with accompanying text:

```
\DeclareDocumentCommand{\LWR@hrefb}{O{} m}{%
    \LWR@ensuredoingapar%
    \LWR@subhyperref{#2}%
    \endgroup% restore catcodes
    \LWR@subhyperreftext%
}

\newrobustcmd*{\LWR@href}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hrefb%
}
```

\LWR@nolinkurl {⟨*URL*⟩}

Print the name of the link without creating the link:

```
\newcommand*{\LWR@nolinkurlb}[1]{%
    \LWR@ensuredoingapar%
    \def\LWR@templink{#1}%
    \@onelevel@sanitize\LWR@templink%
    \LWR@templink%
    \endgroup%
}

\newrobustcmd*{\LWR@nolinkurl}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@nolinkurlb%
}
```

`\LWR@url` {⟨*URL*⟩}

Create a link whose text name is the address of the link.

The url package may redefine `\url`, so it is `\let` to `\LWR@urlahere` and also redefined by lwarp-url.

```
\DeclareDocumentCommand{\LWR@urlb}{m}{%
    \LWR@ensuredoingapar%
    \def\LWR@templink{#1}%
    \@onelevel@sanitize\LWR@templink%
    \LWR@href{\LWR@templink}{\LWR@templink}%
    \endgroup%
}

\newrobustcmd*{\LWR@url}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@urlb%
}
```

`\LWR@subinlineimage` {⟨*1:* <alt> *tag*⟩} {⟨*2: class*⟩} {⟨*3: filename*⟩} {⟨*4: extension*⟩} {⟨*5: css style*⟩} {⟨*6: aria role*⟩}

Factored from `lateximage`.

```
\newcommand*{\LWR@subinlineimage}[6]{%
    \ifblank{#6}%
        {\renewcommand*{\LWR@tempone}{}}%
        {\renewcommand*{\LWR@tempone}{role="#6"\LWR@indentHTML}}%
    \ifblank{#1}%
    {%
        \LWR@htmltag{img \LWR@indentHTML
            src=\textquotedbl#3.#4\textquotedbl \LWR@indentHTML
            alt=\textquotedbl#3\textquotedbl \LWR@indentHTML
            \LWR@tempone
            style=\textquotedbl#5\textquotedbl \LWR@indentHTML
            class=\textquotedbl#2\textquotedbl \LWR@orignewline
        }%
    }%
    {%
        \LWR@htmltag{img \LWR@indentHTML
            src=\textquotedbl#3.#4\textquotedbl \LWR@indentHTML
            alt=\textquotedbl#1\textquotedbl \LWR@indentHTML
            \LWR@tempone
            style=\textquotedbl#5\textquotedbl \LWR@indentHTML
            class=\textquotedbl#2\textquotedbl \LWR@orignewline
        }%
    }%
}
```

```
\end{warpHTML}
```

Table 17: Float data structures

For each <type> of float (figure, table, etc.) there exists the following:

counter <type>: A counter called <type>, such as `figure`, `table`.

\<type>name: Name. `\figurename` prints "Figure", etc.

\ext@<type>: File extension. `\ext@figure` prints "lof", etc.

\fps@<type>: Placement.

\the<type>: Number. `\thetable` prints the number of the table, etc.

\p@<type>: Parent's number. Prints the number of the [within] figure, etc.

\fnum@<type>: Prints the figure number for the caption.
`\<type>name \the<type>`, "Figure 123".

\<type>: Starts the float environment. `\figure` or `\begin{figure}`

\end<type>: Ends the float environment. `\endfigure` or `\end{figure}`

\tf@<ext>: The LaTeX file identifier for the output file.

LWR@have<type>: A boolean remembering whether a `\listof` was requested for a float of this type.

File with extension lo<f,t,a-z>: An output file containing the commands to build the `\listof<type>` "table-of-contents" structure.

Cross-referencing names: For cleveref's `\cref` and related, `\crefname` and `\Crefname` assign human-readable names for references to this float type.

77 Floats

Floats are supported, although partially through emulation.

Table 17 shows the data structure associated with each <type> of float.

77.1 Float environment

for HTML output:
```
\begin{warpHTML}
```

`\LWR@floatbegin {`⟨*type*⟩`} [`⟨*placement*⟩`]` Begins a `\newfloat` environment.

```
\NewDocumentCommand{\LWR@floatbegin}{m o}{%
```

Warn if starting a float inside a :

```
    \LWR@spanwarninvalid{float}%

    \ifbool{FormatWP}{\newline}{}%
    \LWR@stoppars%
```

There is a new float, so increment the unique float counter:

```
    \addtocounter{LWR@thisautoid}{1}%
    \booltrue{LWR@freezethisautoid}%

    \begingroup%
```

Settings while inside the environment:

```
    \LWR@print@raggedright%
```

Open an HTML `figure` tag. The figure is assigned a `class` equal to its type, and another class according to the float package style, if used. Note that `\csuse` returns an empty string if `\LWR@floatstyle@<type>` is not defined.

```
    \LWR@htmltag{%
        figure id=\textquotedbl%
            \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
        \textquotedbl\ % space
        class=\textquotedbl#1 \@nameuse{LWR@floatstyle@#1}\textquotedbl%
    }%
    \ifbool{FormatWP}{%
        \LWR@orignewline%
        \LWR@BlockClassWP{}{}{wp#1}%
    }{}%
```

Update the caption type:

```
    \renewcommand*{\@captype}{#1}%
```

Mark the float for a word processor conversion:

```
    \LWR@startpars%
    \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%

    === begin #1 ===

    }{}%
```

After each `\LWR@floatbegin`, look for `\centering`, etc next, using `\LWR@floatalignment`.

```
}
```

For koma-script. The following does not work for tables.

```
\AtBeginDocument{

\IfPackageLoadedTF{tocbasic}{

\appto\figure@atbegin{%
    \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
}

}{}% tocbasic

}% AtBeginDocument
```

\@xfloat
\@xdlbfloat
Support packages which create floats directly.

Look for \centering, etc using \LWR@floatalignment.

```
\AtBeginDocument{
    \def\@xfloat #1[#2]{%
        \LWR@floatbegin{#1}[#2]
        \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
    }
    \def\@xdblfloat #1[#2]{%
        \LWR@floatbegin{#1}[#2]
        \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
    }
}
```

\LWR@floatend
Ends a \newfloat environment.

```
\newcommand*{\LWR@floatend}{%
```

If saw a \centering, finish the center environment:

```
    \LWR@endfloatalignment%
```

Mark the float end for a word processor conversion:

```
    \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%

    === end ===

    }{}%
    \LWR@stoppars%
```

Close an HTML figure tag:

```
    \ifbool{FormatWP}{\endLWR@BlockClassWP}{}%
    \LWR@htmlelementend{figure}%
    \endgroup%
    \boolfalse{LWR@freezethisautoid}%
    \LWR@startpars%
    \ifbool{FormatWP}{\newline}{}%
}
```

\end@float
\end@dlbfloat
Support packages which create floats directly.

```
\AtBeginDocument{
    \let\end@float\LWR@floatend
    \let\end@dblfloat\LWR@floatend
}
```

77.2 Float tracking

LWR@thisautoid (*Ctr*)
A sequential counter for all floats and theorems. This is used to identify the float or theorem then reference it from the List of Figures and List of Tables.

```
\newcounter{LWR@thisautoid}
```

LWR@thisautoidWP (*Ctr*) A sequential counter for all word processor conversion s. This is used to convince LibreOffice to form a frame around this element.

```
\newcounter{LWR@thisautoidWP}
```

LWR@freezethisautoid (*bool*) Prevents multiple increments of \LWR@thisautoid inside a float.

```
\newbool{LWR@freezethisautoid}
\boolfalse{LWR@freezethisautoid}
```

\LWR@forcenewautoidanchor Adds a new <autoid> anchor.

```
\newcommand*{\LWR@forcenewautoidanchor}{%
    \addtocounter{LWR@thisautoid}{1}%
    \ifbool{LWR@doingapar}%
    {%
        \LWR@htmltag{a id=\textquotedbl%
            \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
            \textquotedbl\ }% space
        \LWR@htmltag{/a }%
    }%
    {%
        \LWR@stoppars%
        \LWR@htmltag{a id=\textquotedbl%
            \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
        \textquotedbl\ }% space
        \LWR@htmltag{/a }%
        \LWR@startpars%
    }%
}
```

\LWR@newautoidanchor Sometimes adds a new <autoid> anchor.

```
\newcommand*{\LWR@newautoidanchor}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {\ifbool{LWR@freezethisautoid}{}{\LWR@forcenewautoidanchor}}%
}
```

\@captype Remembers which float type is in use.

```
\newcommand*{\@captype}{}
```

\LWR@floatalignmentname Set to center, flushleft, or flushright if saw \centering, \raggedright, or \raggedleft.

```
\newcommand*{\LWR@floatalignmentname}{}
```

\LWR@floatalignment If sees a \centering, \raggedleft, or \raggedright, creates a center, flushright, or flushleft environment.

```
\newcommand*{\LWR@floatalignment}{%
    \ifdefstrequal{\LWR@mynexttoken}{\centering}{%
        \center%
        \renewcommand*{\LWR@floatalignmentname}{center}%
    }{}%
```

```
    \ifdefstrequal{\LWR@mynexttoken}{\raggedright}{%
        \flushleft%
        \renewcommand*{\LWR@floatalignmentname}{flushleft}%
    }{}%
    \ifdefstrequal{\LWR@mynexttoken}{\raggedleft}{%
        \flushright%
        \renewcommand*{\LWR@floatalignmentname}{flushright}%
    }{}%
}
```

`\LWR@endfloatalignment` Closes an environment from `\LWR@floatalignment`.

```
\newcommand*{\LWR@endfloatalignment}{%
    \ifdefvoid{\LWR@floatalignmentname}%
        {}%
        {\@nameuse{end\LWR@floatalignmentname}}%
    \renewcommand*{\LWR@floatalignmentname}{}%
}
```

77.3 Caption inside a float environment

`\CaptionSeparator` How to separate the float number and the caption text, if not defined by the user. In most cases, caption's settings are used instead.

```
\AtBeginDocument{\providecommand*{\CaptionSeparator}{:~}}
```

`\@caption` {⟨*posn*⟩} [⟨*name*⟩] {⟨*long name*⟩}

`\@makecaption` {⟨*name and num*⟩} {⟨*text*⟩}

Prints the float type and number, the caption separator, and the caption text.

`\@caption` is provided here in case caption is not loaded, and is based on the nameref package.

```
\AtBeginDocument{
    \IfPackageLoadedTF{caption}{}{
        \let\LWR@orig@caption\@caption%
        \long\def\@caption#1[#2]{%
```

Warn if using a caption inside a :

```
            \LWR@spanwarnformat{caption}%

            \LWR@setlatestname{#2}%
            \LWR@orig@caption{#1}[{#2}]% also takes third argument
        }%

        \renewcommand{\@makecaption}[2]{%
            \LWR@traceinfo{@makecaption}%
            \caption@begin{\@captype}%
            \LWR@isolate{#1}%
            \edef\LWR@tempone{#1}%
            \ifdefvoid{\LWR@tempone}{}{\CaptionSeparator}%
            \LWR@isolate{#2}%
```

```
            \caption@end%
            \LWR@traceinfo{@makecaption: done}%
        }%
    }
}
```

77.4 Caption and LOF linking and tracking

When a new HTML file is marked in the LaTeX PDF file, or at the start of a new section, the LaTeX PDF page number at that point is stored in `LWR@currentautosecfloatpage`, (and the associated filename is remembered by the special LaTeX labels). This page number is used to generate an `autopage` HTML `<id>` in the HTML output at the start of the new HTML file or section. Meanwhile, there is a float counter used to generate an HTML `autoid` `<id>` at the start of the float itself in the HTML file. The `autopage` and `autoid` values to use for each float are written to the `.lof`, etc. files just before each float's entry. These values are used by `\l@figure`, etc. to create the HTML links in the List of Figures, etc.

`LWR@nextautoid` (*Ctr*) Tracks `autoid` for floats. Tracks `autopage` for floats.

`LWR@nextautopage` (*Ctr*) These are updated per float as the `.lof`, `.lot` file is read.

```
\newcounter{LWR@nextautoid}
\newcounter{LWR@nextautopage}
```

`\LWRsetnextfloat` {⟨*autopage*⟩} {⟨*float autoid*⟩}

`*_html.lof` (*file*)
`*_html.lot` (*file*) This is written to the `*_html.lof` or `*_html.lot` file just before each float's usual entry. The `autopage` and the float's `autoid` are remembered for `\l@figure` to use when creating the HTML links.

```
\newcommand*{\LWRsetnextfloat}[2]{%
    \setcounter{LWR@nextautopage}{#1}%
    \setcounter{LWR@nextautoid}{#2}%
}
```

`LWR@figcaption` (*env.*) An HTML `<figcaption>` is not allowed in places where LaTeX does allow a figure caption, such as inside a `longtable` where the tabular has already started, or inside a `center` environment. Therefore, a `<div>` of class `figurecaption` is used instead.

```
\newenvironment*{LWR@figcaption}
    {%
        \ifbool{FormatWP}{%
            \BlockClass[font-style:italic]{figurecaption}%
        }{%
            \BlockClass{figurecaption}%
        }%
```

Inside the caption, temporarily prevent underfull `\hbox` warnings, such as when the caption contains a math SVG image.

```
            \hbadness=10000\relax%
    }%
    {\endBlockClass}
```

\LWR@HTML@caption@begin {⟨*type*⟩}

Low-level code to create HTML tags for captions.

The print versions are from the caption package, if loaded.

```
\newcommand*{\LWR@HTML@caption@begin}[1]
{%
    \LWR@traceinfo{LWR@HTML@caption@begin}%
```

Keep par and minipage changes local:

```
    \begingroup%
```

No need for a minipage or \parbox inside the caption:

```
    \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}{}{}%
    \RenewDocumentCommand{\parbox}{O{t} O{} O{t} m +m}{##5}%
```

Enclose the original caption code inside an HTML tag:

```
    \LWR@figcaption%
    \LWR@traceinfo{LWR@HTML@caption@begin: about to LWR@origcaption@begin}%
    \LWR@print@caption@begin{#1}%
    \LWR@traceinfo{LWR@HTML@caption@begin: done}%
}
```

\LWR@HTML@caption@end Low-level patches to create HTML tags for captions.

```
\newcommand*{\LWR@HTML@caption@end}
{%
    \LWR@traceinfo{LWR@HTML@caption@end}%
    \LWR@print@caption@end%
```

Closing tag:

```
    \endLWR@figcaption%
    \endgroup%
    % \leavevmode% avoid bad space factor (0) error
    \LWR@traceinfo{LWR@HTML@caption@end: done}%
}
```

\caption@begin Low-level patches to create HTML tags for captions. These are assigned \AtBeginDocument
\caption@end so that other packages which modify captions will have already been loaded before saving the print-mode version.

Print versions are provided here in case caption is not loaded.

```
\AtBeginDocument{
    \providecommand{\caption@begin}[1]{}
    \LWR@formatted{caption@begin}

    \providecommand{\caption@end}{}
    \LWR@formatted{caption@end}
}
```

\captionlistentry Tracks the float number for this caption used outside a float. Patched to create an HTML anchor.

```
\AtBeginDocument{%
\IfPackageLoadedTF{caption}{
    \let\LWR@origcaptionlistentry\captionlistentry

    \renewcommand*{\captionlistentry}{%
        \LWR@ensuredoingapar%
        \LWR@origcaptionlistentry%
    }


    \def\LWR@LTcaptionlistentry{%
        \LWR@ensuredoingapar%
        \LWR@forcenewautoidanchor%
        \bgroup%
        \@ifstar{\egroup\LWR@LT@captionlistentry}% gobble *
            {\egroup\LWR@LT@captionlistentry}%
    }%

    \def\LWR@LT@captionlistentry#1{%
        \caption@listentry\@firstoftwo[\LTcaptype]{#1}%
    }%
}% caption loaded
{% caption not loaded
    \newcommand{\captionlistentry}[2][]{}%
    \newcommand{\LWR@LT@captionlistentry}[2][]{}%
}
}% AtBeginDocument
```

\addcontentsline Patched to write the autopage and autoid before each float's entry. No changes if writing .toc For a theorem, automatically defines \ext@<type> as needed, to mimic and reuse the float mechanism.

f

```
\let\LWR@origaddcontentsline\addcontentsline

\renewcommand*{\addcontentsline}[3]{%
    \ifstrequal{#1}{toc}{}{% not TOC


    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {\LWR@newautoidanchor}%


        \ifcsvoid{ext@#2}{\csdef{ext@#2}{#1}}{}%

        \addtocontents{\@nameuse{ext@#2}}{%
            \protect\LWRsetnextfloat%
            {\arabic{LWR@currentautosecfloatpage}}%
            {\arabic{LWR@thisautoid}}%
        }%
    }% not TOC
    \LWR@origaddcontentsline{#1}{#2}{#3}%
}
```

capt-of (*Pkg*) caption (*Pkg*) Either package provides \captionof, which is later patched at the beginning of

the document.

\captionof Patched to handle paragraph tags.

```
\RequirePackage{capt-of}

\AtBeginDocument{
    \let\LWR@origcaptionof\captionof

    \renewcommand*{\captionof}{%
        \LWR@stoppars%
        \LWR@origcaptionof%
    }
}% AtBeginDocument

\end{warpHTML}
```

78 Table of Contents, LOF, LOT

This section controls the generation of the TOC, LOF, and LOT.

The `.toc`, `.lof`, and `.lot` files are named by the source code `\jobname`.

In HTML, the printed tables are placed inside a `<div>` of class `toc`, `lof`, or `lot`.

A "sidetoc" is provided which prints a subset of the TOC on the side of each page other than the homepage.

The regular LaTeX infrastructure is used for TOC, along with some patches to generate HTML output.

for HTML output:

```
\begin{warpHTML}
```

78.1 Reading and printing the TOC

\LWR@myshorttoc {⟨*toc/lof/lot/sidetoc*⟩}

Reads in and prints the TOC/LOF/LOT at the current position. While doing so, makes the `@` character into a normal letter to allow formatting commands in the section names.

Unlike in regular LaTeX, the file is not reset after being read, since the sideTOC may be referred to again in each HTML page.

```
\newcommand*{\LWR@myshorttoc}[1]{%
    \LWR@traceinfo{LWR@myshorttoc: #1}%
```

Only if the file exists:

```
    \IfFileExists{\jobname.#1}{%
        \LWR@traceinfo{LWR@myshorttoc: loading}%
```

⚠ Many of the commands in the file will have `@` characters in them, so `@` must be

made a regular letter.

```
        \begingroup%
        \makeatletter%
```

Disable \ref to avoid nested HTML references.

```
     \LetLtxMacro\ref\LWR@print@ref%
     \LWR@disablepinyin%
```

Read in the TOC file:

```
        \@input{\jobname.#1}%
        \endgroup%
    }%
    {}%
    \LWR@traceinfo{LWR@myshorttoc: done}%
}
```

\LWR@subtableofcontents {⟨*toc/lof/lot*⟩} {⟨*sectionstarname*⟩}

Places a TOC/LOF/LOT at the current position.

```
\NewDocumentCommand{\LWR@subtableofcontents}{m m}{%
```

Closes previous levels:

```
    \@ifundefined{chapter}%
        {\LWR@closeprevious{section}}%
        {\LWR@closeprevious{chapter}}%
```

Prints any pending footnotes so that they appear above the potentially large TOC:

```
    \LWR@printpendingfootnotes%
```

Place the list into its own chapter (if defined) or section:

```
    \@ifundefined{chapter}{\section*{#2}}{\chapter*{#2}}%
```

Create a new HTML nav containing the TOC/LOF/LOT:

```
    \LWR@htmlelementclass{nav}{#1}%
```

Create the actual list:

```
    \LWR@myshorttoc{#1}%
```

Close the nav:

```
    \LWR@htmlelementclassend{nav}{#1}%
}
```

\@starttoc {⟨*ext*⟩}

Patch \@starttoc to encapsulate the TOC inside HTML tags:

```
\let\LWR@orig@starttoc\@starttoc

\renewcommand{\@starttoc}[1]{
    \LWR@htmlelementclass{nav}{#1}%
    \LWR@orig@starttoc{#1}%
    \LWR@htmlelementclassend{nav}{#1}%
}
```

LWR@copiedsidetoc (*bool*) Used to only copy the TOC file to the sidetoc a single time.

(listings and perhaps other packages would re-use `\tableofcontents` for their own purposes, causing the sidetoc to be copied more than once, and thus end up empty.)

```
\newbool{LWR@copiedsidetoc}
\boolfalse{LWR@copiedsidetoc}
```

`\tableofcontents` Patch `\tableofcontents`, etc. to print footnotes first. newfloat uses `\listoffigures` for all future float types.

```
\AtBeginDocument{

\LetLtxMacro\LWR@origtableofcontents\tableofcontents

\renewcommand*{\tableofcontents}{%
```

Do not print the table of contents if formatting for a word processor, which will presumably auto-generate its own updated table of contents:

```
    \ifboolexpr{bool{FormatWP} and bool{WPMarkTOC}}{

    === table of contents ===

    }
    {
```

Copy the `.toc` file to `.sidetoc` for printing the sideTOC. The original `.toc` file is renewed when `\tableofcontents` is finished.

```
        \ifbool{LWR@copiedsidetoc}{}{%
            \LWR@copyfile{\jobname.toc}{\jobname.sidetoc}%
            \booltrue{LWR@copiedsidetoc}%
        }%
        \LWR@printpendingfootnotes
```

Disable `\ref` to avoid nested HTML references.

```
        \begingroup%
        \LetLtxMacro\ref\LWR@print@ref%
        \LWR@disablepinyin%
        \LWR@origtableofcontents%
        \endgroup%
    }
}% \tableofcontents

}% AtBeginDocument
```

\listoffigures

```
\let\LWR@origlistoffigures\listoffigures

\renewcommand*{\listoffigures}{
    \ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{

    === list of figures ===

    }
    {
        \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
        \begingroup%
        \LetLtxMacro\ref\LWR@print@ref%
        \LWR@disablepinyin%
        \LWR@origlistoffigures%
        \endgroup%
    }
}
```

\listoftables

```
\let\LWR@origlistoftables\listoftables

\renewcommand*{\listoftables}{
    \ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{

    === list of tables ===

    }
    {
        \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
        \begingroup%
        \LetLtxMacro\ref\LWR@print@ref%
        \LWR@disablepinyin%
        \LWR@origlistoftables%
        \endgroup%
    }
}
```

78.2 TOC commands

\LWR@listof {⟨*type*⟩} {⟨*title*⟩}

Emulate the \listof command from the float package (section 276). Used to create lists of custom float types. Also used to redefine the standard LaTeX \listoffigures and \listoftables commands, and in tocloft and memoir.

```
\NewDocumentCommand{\LWR@listof}{m +m}{%
    \@ifundefined{l@#1}{%
```

```
    \csdef{l@#1}##1##2{\hypertocfloat{1}{#1}{\@nameuse{ext@#1}}{##1}{##2}}%
  }{}%
  \LWR@subtableofcontents{\@nameuse{ext@#1}}{#2}%
  \expandafter\newwrite\csname tf@\csname ext@#1\endcsname\endcsname%
  \immediate\openout \csname tf@\csname ext@#1\endcsname\endcsname%
      \jobname.\@nameuse{ext@#1}\relax%
}
```

78.3 Side TOC

The "side TOC" is a table-of-contents positioned to the side.

It may be renamed by redefining `\sidetocname`, and may contain paragraphs.

Per table 18, CSS may be used to format the sideTOC.

Table 18: CSS related to the sideTOC

`div.sidetoccontainer`: The entire sideTOC.

`div.sidetoctitle`: The title.

`div.sidetoccontents`: The table of contents.

```
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

`SideTOCDepth` (*Ctr*) Controls how deep the side-TOC gets. Use a standard LaTeX section level similar to `tocdepth`. Warn if parts of the website may be inaccessible.

```
\newcounter{SideTOCDepth}
\setcounter{SideTOCDepth}{1}

\AtEndDocument{%
  \ifnumcomp{\value{SideTOCDepth}}{<}{\value{FileDepth}}{
    \PackageWarningNoLine{lwarp}
    {%
      SideTOCDepth is less than FileDepth,\MessageBreak
      so some website pages may be inaccessible%
    }
  }{}
}
```

`\sidetocname` Holds the default name for the sideTOC.

```
\newcommand{\sidetocname}{Contents}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`\LWR@sidetoc` Creates the actual side-TOC.

```
\newcommand*{\LWR@sidetoc}{%
```

```
    \LWR@forcenewpage
    \LWR@stoppars

```

The entire sideTOC is placed into a nav of class sidetoc.

```
    \LWR@htmlelementclass{div}{sidetoccontainer}
    \LWR@htmlelementclass{nav}{sidetoc}

    \setcounter{tocdepth}{\value{SideTOCDepth}}

```

The title is placed into a of class sidetoctitle, and may contain paragraphs.

```
    \begin{BlockClass}{sidetoctitle}
    \ifcsvoid{thetitle}{}{\InlineClass{sidetocthetitle}{\thetitle}\par}
    \sidetocname
    \end{BlockClass}

```

The table of contents is placed into a of class sidetoccontents.

```
    \begin{BlockClass}{sidetoccontents}
    \LinkHome

    \LWR@myshorttoc{sidetoc}
    \end{BlockClass}
    \LWR@htmlelementclassend{nav}{sidetoc}
    \LWR@htmlelementclassend{div}{sidetoccontainer}
}
```

78.4 Low-level TOC line formatting

`\numberline` {⟨*number*⟩}

(Called from each line in the .aux, .lof files.)

Record this section number for further use:

```
\newcommand*{\LWR@numberline}[1]{%
    \LWR@sectionnumber{#1}\quad%
}

\LetLtxMacro\numberline\LWR@numberline
```

`\LWR@maybetocdata` Replaced by tocdata. Adds author name.

```
\newcommand*{\LWR@maybetocdata}{}
```

`\hypertoc` {⟨*1: depth*⟩} {⟨*2: type*⟩} {⟨*3: name*⟩} {⟨*4: page*⟩}

Called by \l@section, etc. to create a hyperlink to a section.

The autopage label is always created just after the section opens.

#1 is depth

#2 is section, subsection, etc.

#3 the text of the caption

#4 page number

```
\NewDocumentCommand{\hypertoc}{m m +m m}{%
    \LWR@traceinfo{hypertoc !#1!#2!#3!#4!}%
```

Respond to tocdepth:

```
    \ifnumcomp{#1}{>}{\value{tocdepth}}%
        {}%
        {%
            \LWR@startpars%
```

Create an HTML link to <filename>#autosec-(page), with the name, of the given HTML class.

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
            \LWR@subhyperrefclass{%
                \LWR@htmlrefsectionfilename{\BaseJobname-autopage-#4}%
                    \LWR@origpound\LWR@print@mbox{autosec-#4}%
            }{#3}{toc#2}%


            \LWR@maybetocdata%


            \LWR@stoppars%
        }%
    \LWR@traceinfo{hypertoc done}%
}
```

lofdepth (*Ctr*) TOC depth for figures.

```
\IfClassLoadedTF{memoir}{}{
    \newcounter{lofdepth}
    \setcounter{lofdepth}{1}
}
```

lotdepth (*Ctr*) TOC depth for tables.

```
\IfClassLoadedTF{memoir}{}{
    \newcounter{lotdepth}
    \setcounter{lotdepth}{1}
}
```

\hypertocfloat {⟨*1: depth*⟩} {⟨*2: type*⟩} {⟨*3: ext of parent*⟩} {⟨*4: caption*⟩} {⟨*5: page*⟩}

#1 is depth

#2 is figure, table, etc.

#3 is lof, lot, of the parent.

#4 the text of the caption

#5 page number

```
\newcommand{\hypertocfloat}[5]{%
```

If some float-creation package has not yet defined the float type's `lofdepth` counter, etc, define it here:

```
    \@ifundefined{c@#3depth}{%
        \newcounter{#3depth}%
        \setcounter{#3depth}{1}%
    }{}%
```

Respond to `lofdepth`, etc.:

```
    \LWR@traceinfo{hypertocfloat depth is #1 #3depth is \arabic{#3depth}}%
    \ifthenelse{\cnttest{#1}{<=}{\arabic{#3depth}}}%
        {%
            \LWR@startpars%
```

Create an HTML link to filename#autoid-(float number), with text of the caption, of the given HTML class.

`\BaseJobname` is added to the label in case xr or xr-hyper are used.

```
            \LWR@subhyperrefclass{%
            \LWR@htmlrefsectionfilename{%
                \BaseJobname-autopage-\arabic{LWR@nextautopage}%
            }%
            \LWR@origpound\LWR@print@mbox{autoid-\arabic{LWR@nextautoid}}}%
            {#4}{toc#2}%

            \LWR@maybetocdata%

            \LWR@stoppars%
        }%
        {}%
}
```

Automatically called by `\contentsline`:

`\l@book` {⟨*name*⟩} {⟨*page*⟩}

Uses `\DeclareDocumentCommand` in case the class does not happen to have a `\book`.

```
\DeclareDocumentCommand{\l@book}{m m}{\hypertoc{-2}{book}{#1}{#2}}
```

`\l@part` {⟨*name*⟩} {⟨*page*⟩}

Uses `\DeclareDocumentCommand` in case the class does not happen to have a `\part`.

```
\DeclareDocumentCommand{\l@part}{m m}{\hypertoc{-1}{part}{#1}{#2}}
```

\l@chapter {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \chapter.

```
\@ifundefined{chapter}
{}
{
\DeclareDocumentCommand{\l@chapter}{m m}
    {\hypertoc{0}{chapter}{#1}{#2}}
}
```

\l@section {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@section}[2]{\hypertoc{1}{section}{#1}{#2}}
```

\l@subsection {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@subsection}[2]{\hypertoc{2}{subsection}{#1}{#2}}
```

\l@subsubsection {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@subsubsection}[2]{\hypertoc{3}{subsubsection}{#1}{#2}}
```

\l@paragraph {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@paragraph}[2]{\hypertoc{4}{paragraph}{#1}{#2}}
```

\l@subparagraph {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@subparagraph}[2]{\hypertoc{5}{subparagraph}{#1}{#2}}
```

\l@figure {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@figure}[2]{\hypertocfloat{1}{figure}{lof}{#1}{#2}}
```

\l@table {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@table}[2]{\hypertocfloat{1}{table}{lot}{#1}{#2}}
```

```
\end{warpHTML}
```

79 Index and glossary

See:
http://tex.stackexchange.com/questions/187038/how-to-mention-section-number-in-index-created-by-imakeidx

Index links are tracked by the counter LWR@autoindex. This counter is used to create a label for each index entry, and a reference to this label for each entry in

the index listing. This method allows each index entry to link directly to its exact position in the document.

for HTML output:

```
\begin{warpHTML}

\newcounter{LWR@autoindex}
\setcounter{LWR@autoindex}{0}

\newcounter{LWR@autoglossary}
\setcounter{LWR@autoglossary}{0}
```

`\IndexPageSeparator` `\IndexRangeSeparator` User-adjustable delimiters for page and range separators in the `*.ind` files.

```
\newcommand*{\IndexPageSeparator}{, }
\newcommand*{\IndexRangeSeparator}{--}
```

`theindex` (*env.*)

```
\@ifundefined{chapter}
    {\newcommand*{\LWR@indexsection}[1]{\section*{#1}}}
    {\newcommand*{\LWR@indexsection}[1]{\chapter*{#1}}}


\AtBeginDocument{

\renewenvironment*{theindex}{%
    \LWR@indexsection{\indexname}%
    \LetLtxMacro\item\LWR@indexitem%
    \LetLtxMacro\subitem\LWR@indexsubitem%
    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
}{}

}% AtBeginDocument
```

`\LWR@indexitem` `[`⟨*index key*⟩`]` The optional argument is added to support repeatindex.

```
\newcommand{\LWR@indexitem}[1][\@empty]{

    \InlineClass{indexitem}{\LWR@htmlcomment{}}#1%
}
```

`\LWR@indexsubitem`

```
\newcommand{\LWR@indexsubitem}{

    \InlineClass{indexsubitem}{\LWR@htmlcomment{}}%
}
```

`\LWR@indexsubsubitem`

```
\newcommand{\LWR@indexsubsubitem}{

    \InlineClass{indexsubsubitem}{\LWR@htmlcomment{}}%
}
```

`\LWR@xindex@modifyentry` {⟨*indexing term*⟩}

If using *xindex*, modifies the pipe character to become `\hyperindexformat`. The indexing term is split into two argument at the pipe, then fed to `\LWR@xindex@modifyentrysub`.

```
\NewDocumentCommand{\LWR@xindex@modifyentry}{>{\SplitArgument{1}{|}}m}
    {\LWR@xindex@modifyentrysub#1}
```

Handle left and right parenthesis range argument, or add a `hyperindexformat` clause.

```
\newcommand*{\LWR@xindex@modifyentrysub}[2]{%
    \edef\LWR@tempone{#1}%
    \edef\LWR@temptwo{#2}%
    \IfValueTF{#2}{%
        \ifx#2(%
            \appto\LWR@tempone{|(}%
        \else%
            \ifx#2)%
                \appto\LWR@tempone{|)}%
            \else%
                \appto\LWR@tempone{%
                    |hyperindexformat\LWRleftbrace%
                    \LWRbackslash#2%
                    \LWRrightbrace%
                }%
            \fi%
        \fi%
    }%
    {}%
}
```

`LWR@xindex@tricked` (*bool*) Used to track xindex creation. See next.

```
\newbool{LWR@xindex@tricked}
\boolfalse{LWR@xindex@tricked}
```

`\@wrindex` {⟨*indexing term*⟩} Redefined to write the `LWR@autoindex` counter instead of page.

If using *xindex*, the first line is a comment including a special phrase which tricks *xindex* into thinking that hyperref was used.

```
\def\LWR@wrindex#1{%
    \ifbool{LWR@xindex}{%
        \ifbool{LWR@xindex@tricked}{}{%
            \protected@write\@indexfile{}%
            {%
                \LWRpercent\space hyperpage\LWRrightbrace%
                \LWRpercent\space trick xindex to assume hyperref%
            }%
            \global\booltrue{LWR@xindex@tricked}%
        }%
        \LWR@xindex@modifyentry{#1}%
    }{%
        \def\LWR@tempone{#1}%
    }%
    \addtocounter{LWR@autoindex}{1}%
```

```
    \protected@write\@indexfile{}%
{\string\indexentry{\LWR@tempone}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
    \label{LWRindex-\arabic{LWR@autoindex}}%
    \endgroup%
    \@esphack%
}

\AtBeginDocument{
\let\@wrindex\LWR@wrindex
}
```

`\@wrglossary {⟨term⟩}` Redefined to write the `LWR@autoglossary` counter instead of page.

```
\def\@wrglossary#1{%
    \addtocounter{LWR@autoglossary}{1}%
    \LWR@new@label{LWRglossary-\theLWR@autoglossary}%
    \protected@write\@glossaryfile{}%
    {\string\glossaryentry{#1}{\theLWR@autoglossary}}%
    \endgroup%
    \@esphack%
}
```

`\LWR@indexnameref@anonref {⟨LWR@autoindex⟩}`

Displays a reference link where there no `\ref` available.

```
\newcommand*{\LWR@indexnameref@anonref}[1]{%
    \LWR@startref{LWRindex-#1}%
    (*)%
    \LWR@htmltag{/a}%
}
```

`\LWR@indexnameref@ref {⟨LWR@autoindex⟩}`

Creates `\ref`-style index references. To avoid an unwanted space if there is nothing to reference, the reference is checked first.

```
\newcommand*{\LWR@indexnameref@ref}[1]{%
    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
    \ifdefvoid{\LWR@thisref}{}{%
        \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
        \ifdefvoid{\LWR@thisref}%
            {\LWR@indexnameref@anonref{#1}}%
            {\ref{LWRindex-#1}}%
    }%
}
```

`\LWR@indexnameref@refnameref {⟨LWR@autoindex⟩}`

Creates `\ref`-style index references. To avoid an unwanted space if there is nothing to reference, the reference is checked first. For links to starred or ?? objects, only the name is used.

```
\newcommand*{\LWR@indexnameref@refnameref}[1]{%
```

```
    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
    \ifdefvoid{\LWR@thisref}{}{%
        \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
        \ifdefvoid{\LWR@thisref}{}{%
            \ifdefstring{\LWR@thisref}{(*)}%
                {}%
                {\ref{LWRindex-#1} }% space
        }%
    }%
    \nameref{LWRindex-#1}%
}
```

`\LWR@indexnameref@cref {`⟨*LWR@autoindex*⟩`}`

Creates `\cref`-style index references. If no numbered reference is available, a `\nameref` is used instead. If the reference is `??`, which will be changed by `\LWR@indexnameref` to become `(*)`, then the link is changed to show `(*)`.

```
\newcommand*{\LWR@indexnameref@cref}[1]{%
    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
    \ifdefvoid{\LWR@thisref}{%
        \nameref{LWRindex-#1}%
    }{%
        \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
        \ifdefvoid{\LWR@thisref}{%
            \nameref{LWRindex-#1}%
        }{%
            \ifdefstring{\LWR@thisref}{(*)}{%
                \LWR@indexnameref@anonref{#1}%
            }{%
                \cref{LWRindex-#1}%
            }%
        }%
    }%
}
```

`\LWR@indexnameref@crefnameref {`⟨*LWR@autoindex*⟩`}`

Creates `\cref`-style index references. If no numbered reference is available, a `\nameref` is used instead. If the reference is `??`, which will be changed by `\LWR@indexnameref` to become `(*)`, then the link is changed to show only the name.

```
\newcommand*{\LWR@indexnameref@crefnameref}[1]{%
    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
    \ifdefvoid{\LWR@thisref}%
        {}%
        {%
            \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
            \ifdefvoid{\LWR@thisref}%
                {}%
                {%
                    \ifdefstring{\LWR@thisref}{(*)}%
                        {}%
                        {\cref{LWRindex-#1}} % space
                }%
        }%
    \nameref{LWRindex-#1}%
}
```

`\LWR@indexnameref {⟨LWR@autoindex⟩}`

Creates a hyperlink based on the given entry's autoindex.

```
\newcommand*{\LWR@indexnameref}[1]{%
    {% group
```

Temporarily redefine caption's `\caption@xref` because it was printing ?? in the indexes, and also causing error on expansion:

```
        \ifdef{\caption@xref}{%
            \renewcommand*{\caption@xref}[2]{(*)}%
        }{}%

        \ifdefstring{\LWR@IndexRef}{ref}{%
            \LWR@indexnameref@ref{#1}%
        }{%
        \ifdefstring{\LWR@IndexRef}{nameref}{%
            \nameref{LWRindex-#1}%
        }{%
        \ifdefstring{\LWR@IndexRef}{refnameref}{%
            \LWR@indexnameref@refnameref{#1}%
        }{%
        \ifdefstring{\LWR@IndexRef}{cref}{%
            \LWR@indexnameref@cref{#1}%
        }{%
        \ifdefstring{\LWR@IndexRef}{crefnameref}{%
            \LWR@indexnameref@crefnameref{#1}%
        }{%
        \ifdefstring{\LWR@IndexRef}{autoref}{%
            \LWR@indexnameref@cref{#1}%
        }{% text string
            \LWR@startref{LWRindex-#1}%
            \LWR@IndexRef%
            \LWR@htmltag{/a}%
        }}}}}}%
    }% group
}
```

`\LWR@doindexentrysubsub {⟨range start: LWR@autoindex, or macros.⟩} {⟨range end or blank⟩}`

Creates a hyperlink, or handles `\see`, `\textbf`, etc.

```
\newrobustcmd{\LWR@doindexentrysubsub}[2]{%
    \IfInteger{#1}%
        {\LWR@indexnameref{#1}}%
        {#1}%
    \IfValueT{#2}{%
        \IndexRangeSeparator%
        \IfInteger{#2}%
            {\LWR@indexnameref{#2}}%
            {#2}%
    }%
}
```

`\LWR@doindexentrysub {⟨range delimiter⟩} {⟨LWR@autoindex or macros, possible a range⟩}`

```
\NewDocumentCommand{\LWR@doindexentrysub}{m >{\SplitArgument{1}{#1}}m}
```

```
11387     {\LWR@doindexentrysubsub#2}
```

`\LWR@doindexentry` {⟨*LWR@autoindex or macros, possible a range*⟩}

```
11388 \newcommand*{\LWR@doindexentry}[1]{%
11389     \relax% required
11390     \expandafter\LWR@doindexentrysub\expandafter{\IndexRangeSeparator}{#1}%
11391 }
```

`\LWR@hyperindexrefnullified` Handles macros commonly seen inside an `\index` entry. Each macro is redefined to create and format a link to its entry.

⚠ index formatting To handle additional macros:

```
\appto\LWR@hyperindexrefnullified{. . . }
```

```
11392 \newcommand{\LWR@hyperindexrefnullified}{%
11393     \renewrobustcmd{\emph}[1]{\LWR@HTML@emph{\LWR@doindexentry{##1}}}%
11394     \renewrobustcmd{\textbf}[1]{\LWR@HTML@textbf{\LWR@doindexentry{##1}}}%
11395     \renewrobustcmd{\texteb}[1]{\LWR@HTML@texteb{\LWR@doindexentry{##1}}}%
11396     \renewrobustcmd{\textlg}[1]{\LWR@HTML@textlg{\LWR@doindexentry{##1}}}%
11397     \renewrobustcmd{\textrm}[1]{\LWR@HTML@textrm{\LWR@doindexentry{##1}}}%
11398     \renewrobustcmd{\textsf}[1]{\LWR@HTML@textsf{\LWR@doindexentry{##1}}}%
11399     \renewrobustcmd{\texttt}[1]{\LWR@HTML@texttt{\LWR@doindexentry{##1}}}%
11400     \renewrobustcmd{\textup}[1]{\LWR@HTML@textup{\LWR@doindexentry{##1}}}%
11401     \renewrobustcmd{\textsc}[1]{\LWR@HTML@textsc{\LWR@doindexentry{##1}}}%
11402     \renewrobustcmd{\textulc}[1]{\LWR@HTML@textulc{\LWR@doindexentry{##1}}}%
11403     \renewrobustcmd{\textsi}[1]{\LWR@HTML@textsi{\LWR@doindexentry{##1}}}%
11404     \renewrobustcmd{\textit}[1]{\LWR@HTML@textit{\LWR@doindexentry{##1}}}%
11405     \renewrobustcmd{\textsl}[1]{\LWR@HTML@textsl{\LWR@doindexentry{##1}}}%
11406 }
```

`\hyperindexref` {⟨*list of LWR@autoindex, commas, and ranges*⟩}

`\hyperindexref{LWR@autoindex}` is inserted into `*.ind` by the *makeindex* style file `lwarp.ist` or the *xindy* style file `lwarp.xdy`. For *xindex*, `\hyperpage` is inserted, which is `\let` to `\hyperindexref`. For *gindex*, `\addindexitem` and related are inserted, which are defined to use `\hyperindexref`.

The argument is split at commas, and also for ranges, then passed to `\LWR@hyperindexrefsub`.

```
11407 \newcommand*{\hyperindexref}[1]{%
11408     \relax% required
11409     \expandafter\LWR@hyperindexref@comma\expandafter{\IndexPageSeparator}{#1}%
11410 }
```

`\LWR@hyperindexref@comma` {⟨*separator*⟩} {⟨*list of args*⟩}

The list is split at commas, and passed to `\LWR@hyperindexref@@comma`.

```
11411 \NewDocumentCommand{\LWR@hyperindexref@comma}
11412     {m >{\SplitList{#1}} m}
11413     {%
```

Used to place the separtor between each entry, but not before the first.

```
        \def\LWR@hyperindexref@thiscomma{}%
        \def\LWR@hyperindexref@nextcomma{#1}%
```

Each comma-delimited entry is now passed individually to `\LWR@hyperindexref@@comma`.

```
        \ProcessList{#2}\LWR@hyperindexref@@comma%
    }
```

`\LWR@hyperindexref@@comma` {⟨*arg, perhaps with a range*⟩}

A comma separator is placed if not the first item, then the range is parsed.

```
\newcommand*{\LWR@hyperindexref@@comma}[1]{%
    \LWR@hyperindexref@thiscomma%
  \renewcommand{\LWR@hyperindexref@thiscomma}{\LWR@hyperindexref@nextcomma}%
  \expandafter\LWR@hyperindexref@range\expandafter{\IndexRangeSeparator}{#1}%
}
```

`\LWR@hyperindexref@range` {⟨*range delimiter*⟩} {⟨*arg*⟩}

```
\NewDocumentCommand{\LWR@hyperindexref@range}
    {m >{\SplitArgument{1}{#1}} m}
    {\LWR@hyperindexrefsub#2}
```

`\LWR@hyperindexrefsub` {⟨*range start: LWR@autoindex*⟩} {⟨*range end, or* -NoValue-⟩}

Handles the start and end of a range, if applicable.

```
\newcommand*{\LWR@hyperindexrefsub}[2]{%
    \LWR@hyperindexrefsubtwo{#1}%
    \IfValueT{#2}{%
        \IndexRangeSeparator%
        \LWR@hyperindexrefsubtwo{#2}%
    }%
}
```

`\LWR@hyperindexrefsubtwo` {⟨*LWR@autoindex*⟩}

```
\newcommand*{\LWR@hyperindexrefsubtwo}[1]{%
```

In long index lines with numerous entries, *makeindex* can insert a newline before the page number, resulting in an extra space before the first digit. If the first character is a space, remove it first.

```
    \edef\LWR@tempone{#1}%
    \IfBeginWith{\LWR@tempone}{ }{%
        \StrGobbleLeft{\LWR@tempone}{1}[\LWR@tempone]%
    }{}%
```

If a numeric entry, create a link. If not numeric, such as `\see`, use the entry as-is. `\emph`, `\textit`, etc. have been redefined above to create and format the entry.

```
    \IfInteger{\LWR@tempone}%
        {\LWR@indexnameref{\LWR@tempone}}%
        {%
            \begingroup%
```

```
            \LWR@hyperindexrefnullified%
            #1%
            \endgroup%
        }%
}
```

`\hyperpage` Emulate hyperref.

```
\LetLtxMacro\hyperpage\hyperindexref
```

`\nohyperpage` Emulate hyperref.

```
\def\nohyperpage#1{}
```

`\hyperindexformat` Emulate hyperref.

```
\def\hyperindexformat#1#2{%
    #1{\hyperpage{#2}}%
}%
```

```
\end{warpHTML}
```

for PRINT output: A null command for print mode, in case hyperref was not used:

```
\begin{warpprint}
\newcommand{\hyperindexref}[1]{#1}
\end{warpprint}
```

for HTML & PRINT: For the glossaries package, try to prevent an error where `\glo@name` was not found:

```
\begin{warpall}
\providecommand{\glo@name}{}
\end{warpall}
```

80 Bibliography presentation

for HTML output:

```
\begin{warpHTML}
```

`\bibliography` {⟨*filenames*⟩} At one time this was modified to read `\BaseJobname.bbl`, which meant the HTML version could not resolve until the print version was also present. This also confused multibib. It has been reverted to the original to use `\jobname.bbl`.

`\@biblabel` {⟨*text-refnumber*⟩}

```
\renewcommand{\@biblabel}[1]{[#1]\quad}
```

`thebibliography` (*env.*) To emphasize document titles in the bibliography, the following redefines `\em` inside `thebibliography` to gather everything until the next closing brace, then display these tokens with `\textit`.

Adapted from embracedef.sty, *which is by* Takayuki YATO*:*
https://gist.github.com/zr-tex8r/b72555e3e7ad2f0a37f1

```
\AtBeginDocument{

\AtBeginEnvironment{thebibliography}{

\providecommand*{\LWR@newem}[1]{\textit{#1}}

\renewrobustcmd{\em}{%
  \begingroup
    \gdef\LWR@em@after{\LWR@em@finish\LWR@newem}%
    \afterassignment\LWR@em@after
    \toks@\bgroup
}

\def\LWR@em@finish#1{%
    \xdef\LWR@em@after{\noexpand#1{\the\toks@}}%
  \endgroup
  \LWR@em@after\egroup
}

}% \AtBeginEnvironment{thebibliography}

}% \AtBeginDocument

\end{warpHTML}
```

81 Restoring original formatting

for HTML output!

```
\begin{warpHTML}
```

\LWR@restoreMathJaxformatting A few macros (ref: tcolorbox) must be treated separately while printing the HTML comment for a MathJax expression. These are set here, to which other functions may be appended.

```
\newcommand*{\LWR@restoreMathJaxformatting}{}
```

\LWR@restoreorigformatting Used to temporarily restore the print-mode meaning of a number of formatting, graphics, and symbols-related macros while generating SVG math or a lateximage.

Must be used inside a group.

Sets \LWR@formatting to print until the end of the group.

A number of packages will \appto additional actions to this macro.

Various packages add to this macro using \appto.

```
\newcommand*{\LWR@restoreorigformatting}{%
    \LWR@traceinfo{LWR@restoreorigformatting}%
```

Numerous macros change their print/HTML meaning depending on \LWR@formatting:

```
    \renewcommand*{\LWR@formatting}{print}%
    \linespread{1}%
```

```
    \setbool{LWR@doingparhooks}{false}%

    \def\color@endgroup{\endgraf\endgroup}%

    \LetLtxMacro\hfil\LWR@orighfil%
    \let\hss\LWR@orighss%
    \let\llap\LWR@origllap%
    \let\rlap\LWR@origrlap%
    \let\hfilneg\LWR@orighfilneg%

    \let\,\LWR@origcomma% disable HTML short unbreakable space
    \let\textless\LWR@origtextless%
    \let\textgreater\LWR@origtextgreater%

    \let\&\LWR@origampersand%

    \LetLtxMacro\em\LWR@origem%
    \LetLtxMacro\normalfont\LWR@orignormalfont%
    \let\sp\LWR@origsp%
    \let\sb\LWR@origsb%
    \LetLtxMacro\underline\LWR@origunderline%
    \let~\LWR@origtilde%
```

\endtabular must be restored to its original, instead of relying on lwarp's \LWR@formatted mechanism:

```
    \LetLtxMacro\endtabular\LWR@origendtabular%
    \csletcs{endtabular*}{LWR@origendtabular*}%

    \LetLtxMacro\noalign\LWR@orignoalign%
    \LetLtxMacro\hline\LWR@orighline%

    \let\newline\LWR@orignewline%
    \LetLtxMacro\includegraphics\LWR@origincludegraphics%

    \LetLtxMacro\@ensuredmath\LWR@origensuredmath%

    \let\math\LWR@orig@math%
    \let\endmath\LWR@orig@endmath%
    \let\displaymath\LWR@orig@displaymath%
    \let\enddisplaymath\LWR@orig@enddisplaymath%
%
    \LWR@restoreorigaccents%
    \LWR@restoreoriglists%

    \let\@mpfootnotetext\LWR@orig@mpfootnotetext%

    \LWR@hook@processingtags%
```

To enable MathJax-specific nullification, used for tcolorbox:

```
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
        {\LWR@restoreMathJaxformatting}%
        {}%
}
```

```
\end{warpHTML}
```

82 Nullifying filename formatting

The following are used to nullify certain macros and environments while converting section names to file names.

for HTML output!

```
\begin{warpHTML}
```

Also commonly used are `\@empty`, `\@gobble`, and `\@firstofone`.

```
\newcommand*{\LWR@dash}{-}
```

`\LWR@nullfonts` Removes formatting during filename operations, file references, and HTML comments.

⚠ Use only inside a group.

The following are *not* made robust, since they must be expanded to their nullified versions.

```
\catcode`\$=\active% redefining $ below
\catcode`\_=12% redefining \_ below
\newcommand*{\LWR@nullfonts}{%
```

Various built-in symbols.

```
    \renewcommand*{\$}{-}%
    \renewcommand*{\%}{-}%
    \renewcommand*{\_}{-}%
    \renewcommand*{\}}{-}%
    \renewcommand*{\{}{-}%
    \renewcommand*{\&}{-}% used to be 'and'
    \renewcommand*{\#}{-}%
    \renewcommand*{\,}{-}%
    \renewcommand*{~}{-}%
%
% accents:
    \renewcommand*{\`}[1]{##1}%
    \renewcommand*{\'}[1]{##1}%
    \renewcommand*{\^}[1]{##1}%
    \renewcommand*{\~}[1]{##1}%
    \renewcommand*{\=}[1]{##1}%
    \renewcommand*{\u}[1]{##1}%
    \renewcommand*{\.}[1]{##1}%
    \renewcommand*{\"}[1]{##1}%
    \renewcommand*{\H}[1]{##1}%
    \renewcommand*{\v}[1]{##1}%
    \renewcommand*{\d}[1]{##1}%
    \renewcommand*{\c}[1]{##1}%
    \renewcommand*{\b}[1]{##1}%
    \renewcommand*{\t}[1]{##1}%
%
    \let\newline\LWR@dash%
    \let\textasciicircum\LWR@dash%
```

```
    \let\textasciitilde\LWR@dash%
    \let\textasteriskcentered\LWR@dash%
    \let\textbackslash\LWR@dash%
    \let\textbar\LWR@dash%
    \let\textbardbl\LWR@dash%
    \let\textbigcircle\LWR@dash%
    \let\textbraceleft\LWR@dash%
    \let\textbraceright\LWR@dash%
    \let\textbullet\LWR@dash%
    \let\textcopyright\LWR@dash%
    \let\textdagger\LWR@dash%
    \let\textdaggerdbl\LWR@dash%
    \let\textdollar\LWR@dash%
    \let\textellipsis\LWR@dash%
    \let\textemdash\LWR@dash%
    \let\textendash\LWR@dash%
    \let\textexclamdown\LWR@dash%
    \let\textgreater\LWR@dash%
    \let\textless\LWR@dash%
    \let\textordfeminine\LWR@dash%
    \let\textordmasculine\LWR@dash%
    \let\textparagraph\LWR@dash%
    \let\textperiodcentered\LWR@dash%
    \let\textpertenthousand\LWR@dash%
    \let\textperthousand\LWR@dash%
    \let\textquestiondown\LWR@dash%
    \let\textquotedblleft\LWR@dash%
    \let\textquotedblright\LWR@dash%
    \let\textquoteleft\LWR@dash%
    \let\textquoteright\LWR@dash%
    \let\textregistered\LWR@dash%
    \let\textsection\LWR@dash%
    \let\textsterling\LWR@dash%
    \let\texttrademark\LWR@dash%
    \let\textunderscore\LWR@dash%
    \let\textvisiblespace\LWR@dash%
    \let\copyright\LWR@dash%
    \let\dag\LWR@dash%
    \let\ddag\LWR@dash%
    \let\dots\LWR@dash%
    \let\P\LWR@dash%
    \let\pounds\LWR@dash%
    \let\S\LWR@dash%
%
    \renewcommand*{\aa}{a}%
    \renewcommand*{\AA}{A}%
    \renewcommand*{\AE}{AE}%
    \renewcommand*{\ae}{ae}%
    \renewcommand*{\dh}{d}%
    \renewcommand*{\DH}{D}%
    \renewcommand*{\DJ}{D}%
    \renewcommand*{\dj}{d}%
    \renewcommand*{\IJ}{IJ}%
    \renewcommand*{\ij}{ij}%
    \renewcommand*{\L}{L}%
    \renewcommand*{\l}{l}%
    \renewcommand*{\NG}{NG}%
    \renewcommand*{\ng}{ng}%
    \renewcommand*{\O}{O}%
    \renewcommand*{\o}{o}%
```

```
    \renewcommand*{\oe}{oe}%
    \renewcommand*{\OE}{OE}%
    \renewcommand*{\ss}{ss}%
    \renewcommand*{\SS}{SS}%
    \renewcommand*{\th}{th}%
    \renewcommand*{\TH}{TH}%
%
    \let\guillemotleft\@empty%
    \let\guilsinglleft\@empty%
    \let\quotedblbase\@empty%
    \let\textquotedbl\@empty%
    \let\guillemotright\@empty%
    \let\guilsinglright\@empty%
    \let\quotesinglbase\@empty%

    \renewcommand*{\HTMLunicode}[1]{}%
    \renewcommand*{\HTMLentity}[1]{}%

    \renewcommand{\textsuperscript}[1]{##1}%
    \renewcommand{\textsubscript}[1]{##1}%


    \renewcommand{\underline}[1]{##1}%


    \RenewDocumentCommand{\hspace}{s m}{}%


    \RenewDocumentCommand{\LWR@htmlspanclass}{o D(){} m +m}{{##4}}%
   \DeclareExpandableDocumentCommand{\InlineClass}{D{(}{)}{} o m +m}{{##4}}%
```

Nullify math macros.

```
    \def\(##1\){}%
    \def\[##1\]{}%
    \RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
```

Nullify logos:

```
    \renewcommand*{\TeX}{TeX}%
    \renewcommand*{\LaTeX}{LaTeX}%
    \renewcommand*{\LaTeXe}{LaTeX2e}%
    \renewcommand*{\LuaTeX}{LuaTeX}%
    \renewcommand*{\LuaLaTeX}{LuaLaTeX}%
    \renewcommand*{\XeTeX}{XeTeX}%
    \renewcommand*{\XeLaTeX}{XeLaTeX}%
    \renewcommand*{\ConTeXt}{ConTeXt}%
    \renewcommand*{\BibTeX}{BibTeX}%
    \renewcommand*{\MakeIndex}{MakeIndex}%
    \renewcommand*{\AmS}{AmS}%
    \renewcommand*{\MiKTeX}{MiKTeX}%
    \renewcommand*{\LyX}{LyX}%
```

Use the simpler form with `\texorpdfstring`:

```
    \def\texorpdfstring{\expandafter\@secondoftwo}%
}
\catcode`\$=3%
\catcode`\_=8%
```

`\FilenameNullify` {⟨*redefinitions*⟩}

Adds more nullifying definitions for filename generation.

```
\newcommand*{\FilenameNullify}[1]{%
    \appto{\LWR@nullfonts}{#1}%
}
```

```
\end{warpHTML}
```

83 Math

83.1 Limitations

See Math, section 8.7.

83.2 HTML `alt` tag names

Redefinable names for the HTML `alt` tags, for translation according to the reader's native language.

for HTML & PRINT?

```
\begin{warpall}
```

`\AltTextOpen` The opening part of HTML `alt` tag for an image. The default is a left parenthesis.
Default: `(`

```
\newcommand*{\AltTextOpen}{(}
```

`\AltTextClose` The closing part of HTML `alt` tag for an image. The default is a right parenthesis.
Default: `(`

```
\newcommand*{\AltTextClose}{)}
```

`\ImageAltText` The HTML `alt` tag for an image.
Default: `image`

```
\newcommand*{\ImageAltText}{image}
```

`\MathImageAltText` The HTML `alt` tag for an SVG math image.
Default: “`math image`”

```
\newcommand*{\MathImageAltText}{math image}
```

`\LWR@ThisAltText` The HTML `alt` tag for the next image. Cleared after use, and also after each `lateximage`, `\LWR@subsingledollar`, and each use of MathJax.

```
\newcommand*{\LWR@ThisAltText}{}
```

\ThisAltText {⟨*text*⟩}

Assigns the HTML `alt` tag for the next image generated by lwarp, such as a `lateximage`, `picture`, or SVG math.

```
\newcommand*{\ThisAltText}[1]{%
    \renewcommand{\LWR@ThisAltText}{#1}%
}
```

\PackageDiagramAltText
Default: "diagram"

Appended to the `lateximage` HTML `alt` tag for the images generated by many packages.

```
\newcommand*{\PackageDiagramAltText}{diagram}
```

```
\end{warpall}
```

83.3 Inline and display math

for HTML output:

```
\begin{warpHTML}
```

LWR@externalfilecnt (*Ctr*)

Counter for the external files which are generated and then referenced from the HTML:

```
\newcounter{LWR@externalfilecnt}
```

LWR@indisplaymathimage (*bool*)

True if processing display math for SVG output. Inside a `lateximage`, display math is only set to print-mode output if `LWR@indisplaymathimage` is false. Used to avoid nullifying display math before it has been completed.

```
\newbool{LWR@indisplaymathimage}
```

LWR@insidemathcomment (*bool*)

True while inside an HTML comment which is displaying a math environment. Used to undo the comment for a moment while creating a `\label`, so that the label's HTML tags will be seen by HTML.

```
\newbool{LWR@insidemathcomment}
\boolfalse{LWR@insidemathcomment}
```

LWR@xfakebold (*bool*)

True if xfakebold `\setBold` is in use.

```
\newbool{LWR@xfakebold}
\boolfalse{LWR@xfakebold}
```

\LWR@orig@setBold

Redefined by lwarp-xfakebold.

```
\newcommand*{\LWR@orig@setBold}{}
```

\LWR@orig@unsetBold

Redefined by lwarp-xfakebold.

```
\newcommand*{\LWR@orig@unsetBold}{}
```

\LWR@applyxfakebold

Redefined by lwarp-xfakebold.

```
\newcommand*{\LWR@applyxfakebold}{}
```

\LWR@setcurrentfont Sets the actual LaTeX font to that which was selected for HTML output. Ex: In HTML mode, \bfseries sets \LWR@f@series to "bf". This sets the PDF output here for use inside a lateximage.

```
\newcommand*{\LWR@setcurrentfont}{%
    \LWR@traceinfo{Using font family \LWR@f@family}%
    \@nameuse{LWR@print@\LWR@f@family family}%
    \LWR@traceinfo{Using font series \LWR@f@series}%
    \@nameuse{LWR@print@\LWR@f@series series}%
    \LWR@traceinfo{Using font shape \LWR@f@shape}%
    \@nameuse{LWR@print@\LWR@f@shape shape}%
    \LWR@traceinfo{Using font caps shape \LWR@f@shapecaps}%
    \@nameuse{LWR@print@\LWR@f@shapecaps shape}%
}
```

\$ Plain dollar signs appearing in the HTML output may be interpreted by MathJax to be math shifts. For a plain text dollar \$, use an HTML entity to avoid it being interpreted by MathJax, unless are inside a lateximage, in which case it will not be seen by MathJax.

```
\let\LWR@origtextdollar\$

\renewcommand*{\$}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {\LWR@origtextdollar}%
        {\HTMLunicode{00024}}%
}
```

lwarp_baseline_marker.png (*file*)
lwarp_baseline_marker.eps (*file*)

A marker to be used to help *pdfcrop* identify the inline math baseline and width. If either graphicx or graphics is loaded, this marker is placed at the lower left and lower right corners of the inline math. *pdfcrop* is then able to identify the width of the image, and also the height of an image such as a horizontal dash which does not otherwise touch the baseline.

A marker with alpha or opacity of 0% is not registered by *pdfcrop*, so the marker is a small square block of 1% alpha, which seems to work while still being effectively invisible in the final SVG image.

If graphicx is loaded, this marker is sized as a tiny 1 sp square. If graphics is loaded, this marker is used at its default size of around .25 pt. If neither graphics package is loaded, the marker is replaced by a 10 sp horizontal space, and there is no assistance for determining baseline or width of the inline math image. The best results are obtained when using graphicx.

\LWR@addbaselinemarker Places a small marker in an SVG inline image. If graphics or graphicx are loaded, the marker is a mostly transparent image. If neither is loaded, no marker is used.

```
\AtBeginDocument{

\ifpdf
    \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
\else
    \ifXeTeX
        \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
    \else
        \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.eps}
```

```
    \fi
\fi

\IfFileExists{\LWR@baselinename}%
{
    \IfPackageLoadedTF{graphicx}{
        \newcommand*{\LWR@addbaselinemarker}{%
            \LWR@origincludegraphics{\LWR@baselinename}%
        }
    }{
        \IfPackageLoadedTF{graphics}{
            \newcommand*{\LWR@addbaselinemarker}{%
                \LWR@origincludegraphics{\LWR@baselinename}%
            }
        }{
            \newcommand*{\LWR@addbaselinemarker}{%
                \global\booltrue{LWR@warnbaselinemarker}%
            }
            \AtEndDocument{
                \ifbool{LWR@warnbaselinemarker}{
                    \PackageNoteNoLine{lwarp}{%
                        Load graphicx or graphics for improved\MessageBreak
                        SVG math sizing and baselines%
                    }
                }{}
            }
        }
    }
}{% lwarp_baseline_marker.png or .eps is not present
    \newcommand*{\LWR@addbaselinemarker}{%
        \global\booltrue{LWR@warnbaselinemarker}%
    }
    \AtEndDocument{
        \ifbool{LWR@warnbaselinemarker}{
            \PackageWarningNoLine{lwarp}{%
                File \LWR@baselinename\space is not installed\MessageBreak
                alongside the lwarp-*.sty files, so\MessageBreak
                SVG math sizing and baselines may not be accurate}
        }{}
    }
}

}% AtBeginDocument
```

`LWR@warnbaselinemarker` (*bool*) True if the math baseline marker was ever called for, but graphics or graphicx were not loaded.

```
\newbool{LWR@warnbaselinemarker}
\boolfalse{LWR@warnbaselinemarker}
```

`LWR@unknownmathsize` (*bool*) If Ti*k*Z or other objects are used inside math mode, the resulting image may exceed the TEX box, resulting in an incorrect measurement of the size of the resulting image. If this is so, the HTML styles for image size and depth will be neutralized.

```
\newbool{LWR@unknownmathsize}
```

`\LWR@singledollarmeasure` {⟨*math expression*⟩}

Measures the size of the image of the math expression.

(In some circumstances SVG math is used even if MATHJAX is preferred.)

SVG math: \LWR@origensuredmath is part of argument #4.

SVG math \ensuremath: \LWR@origensuredmath is part of argument #4.

SVG dynamic math: \LWR@origensuredmath is part of argument #4.

MATHJAX: Argument #4 is the contents of the math expression without \LWR@origensuredmath. This case is handled above.

MATHJAX \ensuremath: \LWR@origensuredmath is part of argument #4.

MATHJAX dynamic math: Argument #4 is the contents of the math expression without \LWR@origensuredmath, so \LWR@origensuredmath is added below.

\ifmmode: Included "just in case".

Factored from \LWR@subsingledollarsvg.

```
\newcommand*{\LWR@singledollarmeasure}[1]{%
    \begingroup%
```

Temporarily disable formatting while measuring the image parameters:

```
    \LWR@restoreorigformatting%
    \RenewDocumentEnvironment{lateximage}{s o s o o d()}{}{}% inside group
    \LWR@print@normalsize%
```

Temporarily set font for the HTML PDF output:

```
    \LWR@setcurrentfont%
```

lateximagedepth must be nested to avoid generating paragraph tags. $\mathcal{AMS}$ math modifies the \text macro such that \addtocounter does not always occur as expected. Lower-level code is used instead.

```
    \global\advance\c@LWR@lateximagedepth 1\relax%
```

Typeset the math in a box. While doing so, some macros or environments may set LWR@unknownmathsize, in which case this will be used to cancel the HTML styles being generated here.

```
    \boolfalse{LWR@unknownmathsize}%
    \ifmmode%
        \global\sbox{\LWR@singledollarbox}{#1}%
    \else%
        \ifbool{LWR@dynamicmath}{%
            \ifbool{mathjax}{%
                \global\sbox{\LWR@singledollarbox}%
                    {\LWR@origensuredmath{#1}}%
            }{%
                \global\sbox{\LWR@singledollarbox}{#1}%
            }%
        }{%
            \global\sbox{\LWR@singledollarbox}{#1}%
        }%
    \fi%
```

Add a small and almost transparent marker at the depth of the image.

A math minus sign has the same depth as a plus, even though it does not draw anything below the baseline. This means that *pdfcrop* would crop the image without depth. The marker below the baseline is seen by *pdfcrop* and preserves the depth.

```
    \global\sbox{\LWR@singledollarbox}{%
        \usebox{\LWR@singledollarbox}%
        \raisebox{-\dp\LWR@singledollarbox}{%
            \LWR@addbaselinemarker%
        }%
    }%
```

More low-level code to undo the counter change.

```
    \global\advance\c@LWR@lateximagedepth -1\relax% Due to AmS \text macro.
```

Measure the depth:

```
    \setlength{\LWR@singledollardepth}{%
        \LateximageFontScale\dp\LWR@singledollarbox%
    }%
```

Make the length a global change:

```
    \global\LWR@singledollardepth=\LWR@singledollardepth%
```

Likewise for width:

```
    \setlength{\LWR@singledollarwidth}{%
        \LateximageFontScale\wd\LWR@singledollarbox%
    }%
    \global\LWR@singledollarwidth=\LWR@singledollarwidth%
```

Likewise for total height:

```
    \setlength{\LWR@singledollarheight}{%
        \LateximageFontScale\ht\LWR@singledollarbox%
    }%
    \addtolength{\LWR@singledollarheight}{%
        \LateximageFontScale\dp\LWR@singledollarbox%
    }%
    \global\LWR@singledollarheight=\LWR@singledollarheight%

    \endgroup%
}
```

`\LWR@subsingledollarsvg` * {⟨*2: alt text*⟩} {⟨*3: add'l hashing*⟩} {⟨*4: math expression*⟩}

For inline math. Uses SVG math. The image is measured and ajusted to the baseline of the HTML output, and placed inside a `lateximage`.

(In some circumstances SVG math is used even if MathJax is preferred.)

Factored from `\LWR@subsingledollar`.

```
\newcommand*{\LWR@subsingledollarsvg}[4]{%
    \LWR@traceinfo{LWR@subsingledollartsvg}%
```

Measure the depth, width, and height of the math image:

```
    \LWR@singledollarmeasure{#4}%
```

Set a style for the the height or width. The em unit is used so that the math scales according to the user's selected font size.

Start with the greater of the width or the height, biased towards the width:

```
    \ifdimgreater{\LWR@singledollarwidth}{.7\LWR@singledollarheight}{%
        \def\LWR@singledollarstyle{%
            width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
        }%
    }{%
        \def\LWR@singledollarstyle{%
            height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
        }%
    }%
```

If a very narrow width, use the height.

```
    \ifdimless{\LWR@singledollarwidth}{.2em}%
    {%
        \def\LWR@singledollarstyle{%
            height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
        }%
    }%
    {}%
```

If very wide and short, use the width:

```
    \ifdimless{\LWR@singledollarheight}{.2em}%
    {%
        \def\LWR@singledollarstyle{%
            width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
        }%
    }%
    {}%
```

If there is significant text depth, add the depth to the style.

```
    \ifdimgreater{\LWR@singledollardepth}{0.05ex}{%
        \def\LWR@singledollardepthstyle{%
            \ ; % extra space
            \LWR@print@mbox{%
           vertical-align:-\LWR@convertto{em}{\the\LWR@singledollardepth} em%
            } % extra space
        }%
    }{%
        \def\LWR@singledollardepthstyle{}%
    }%
```

If using certain Ti*k*Z actions inside math, the resulting image may exceed the TEX boundaries, so the HTML size styles may be incorrect, and must be neutralized.

```
    \ifbool{LWR@unknownmathsize}{%
        \def\LWR@singledollarstyle{}%
        \def\LWR@singledollardepthstyle{}%
    }{}%
```

Create the `lateximage` using the alternate tag and the computed size and depth. The star causes `lateximage` to use an MD5 hash as the filename. When hashing, also include the current font and color in the hash.

```
    \ifbool{LWR@dynamicmath}{%
        \LWR@traceinfo{subsingledollarsvg: dynamic}%
        \begin{lateximage}% no hashing
            [\MathImageAltText]% alt tag
            []% no add'l hashing
            [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
            (math)% ARIA
    }{% not dynamic math
        \LWR@traceinfo{subsingledollarsvg: static}%
        \IfValueTF{#1}{% #1 True
            \LWR@findcurrenttextcolor% sets \LWR@tempcolor
```

Support for xfakebold:

```
            \ifbool{LWR@xfakebold}%
                {\def\LWR@tempone{Y}}%
                {\def\LWR@tempone{N}}%

            \LWR@traceinfo{subsingledollarsvg about to lateximage}%
            \begin{lateximage}*% use hashing
                [#2]% alt
                *% do not add open/closing braces
                [% addl' hashing
                    #3%
                    FM\LWR@f@family%
                    SR\LWR@f@series%
                    SH\LWR@f@shape%
                    SHC\LWR@f@shapecaps%
                    CL\LWR@tempcolor%
                    FB\LWR@tempone% xfakebold
                ]%
                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
                (math)% ARIA
                \LWR@traceinfo{subsingledollar did lateximage}%
        }{% #1 False
            \begin{lateximage}% no hashing
                [#2]% alt
                []% no add'l hashing
                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
                (math)% ARIA
        }%
    }% not dynamic math
```

Place small and almost transparent markers on the baseline at the left and right edges of the image. These markers are seen by *pdfcrop*, and force vertically-centered objects such as a dash to be raised off the baseline in the cropped image, and also force the total width and left / right margins to be correct. (Except that in some fonts a character may exceed the bounding box, and thus may appear wider than expected when converted to an image.)

```
11884     \LWR@addbaselinemarker%
```

Support for xfakebold:

```
11885     \LWR@applyxfakebold%
```

Typeset the contents:

```
11886     \usebox{\LWR@singledollarbox}%
```

The closing baseline marker:

```
11887     \LWR@addbaselinemarker%

11888     \end{lateximage}%
11889 %
11890 }
```

`\LWR@subsingledollar` * {⟨*2: alt text*⟩} {⟨*3: add'l hashing*⟩} {⟨*4: math expression*⟩}

For inline math. Uses MathJax, or for svg math the image is measured and ajusted to the baseline of the html output, and placed inside a `lateximage`.

svg math: `\LWR@origensuredmath` is part of argument #4.

svg math `\ensuremath`: `\LWR@origensuredmath` is part of argument #4.

svg dynamic math: `\LWR@origensuredmath` is part of argument #4.

MathJax: Argument #4 is the contents of the math expression without `\LWR@origensuredmath`. This case is handled above.

MathJax `\ensuremath`: `\LWR@origensuredmath` is part of argument #4.

MathJax dynamic math: Argument #4 is the contents of the math expression without `\LWR@origensuredmath`, so `\LWR@origensuredmath` is added below.

image filename hashing

If starred, a hashed filename is used.If so, the hash is based on the `alt` tag and also the additional hashing argument.

This may be used to provide an expression with a simple `alt` tag but also enough additional information to provide a unique hash.

An example is when the expression is a complicated TEX expression, which would not copy/paste well. A simplified tag may be used, while the complicated expression is used in the additional hashing argument to ensure a unique image.

Another example is when the expression is simple, but the image depends on options. These options may be decoded into text form and included in the additional hashing argument in order to make the hash unique according to the set of options, even if the simple `alt` tag is still the same.

```
11891 \newlength{\LWR@singledollarwidth}
11892 \newlength{\LWR@singledollarheight}
11893 \newlength{\LWR@singledollardepth}
11894
11895 \newsavebox{\LWR@singledollarbox}
```

```
11896
11897 \NewDocumentCommand{\LWR@subsingledollar}{s m m m}{%
11898     \LWR@traceinfo{LWR@subsingledollar !#2!}%

11899     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
11900     {%
11901         \LWR@traceinfo{LWR@subsingledollar: already in a lateximage}%
11902             #4% contents
11903     }%
11904     {% not in a lateximage
11905         \begingroup%
```

Support for xfakebold:

```
11906         \LWR@applyxfakebold%
```

MathJax cannot parse the often complicated TEX expressions which appear in the various uses of `\ensuredmath`. `\ensuremath` forces the `alt` tag to "`(math image)`", as translated according to `\MathImageAltText`. If this is the case, force the use of a `lateximage` even if MathJax. Likewise for siunitx if `parse-numbers=false`.

If MathJax, or if formatting math for a word processor, and not `\ensuredmath`, and not a dynamic math expression, print the math expression:

```
11907         \ifboolexpr{%
11908             (
11909                 bool{mathjax} or
11910                 ( bool{FormatWP} and bool{WPMarkMath} )
11911             ) and
11912             ( not test {
11913                     \ifstrequal {#2}% from \ensuredmath
11914                         {\AltTextOpen\MathImageAltText\AltTextClose}
11915                 }
11916             ) and
11917             ( not bool{LWR@dynamicmath} )
11918         }%
```

For MathJax, print the math between `\(` and `\)`:

```
11919         {%
11920             \LWR@traceinfo{LWR@subsingledollar: Mathjax}%
11921             {%
11922                 \textbackslash(%
11923                 {%
```

`\ifmmode` to avoid error about `\ttfamily` inside math mode in the case of nested math, ex. equation with tcolorbox with math.

```
11924                     \ifmmode\else\LWR@print@ttfamily\fi%
11925                     \LWR@HTMLsanitizedetokenized{\detokenize{#4}}%
11926                 }%
11927                 \textbackslash)%
11928             }%
11929         }% mathjax
```

For SVG, print the math inside a `lateximage`, with an `<alt>` tag of the LaTeX code, and a css style to control the baseline adjustment.

```
        {% not mathjax
            \LWR@traceinfo{%
           LWR@subsingledollar: NOT mathjax, or is ensuremath, or is dynamic%
            }%
            \LWR@subsingledollarsvg{#1}{#2}{#3}{#4}%
        }% not mathjax
        \endgroup%
    }% not in a lateximage
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
    \LWR@traceinfo{LWR@subsingledollar: done}%
}

\LetLtxMacro\LWR@origdollar$
\LetLtxMacro\LWR@secondorigdollar$% balance for editor syntax highlighting

\LetLtxMacro\LWR@origopenparen\(
\LetLtxMacro\LWR@origcloseparen\)
\LetLtxMacro\LWR@origopenbracket\[
\LetLtxMacro\LWR@origclosebracket\]
```

`$` Redefine the dollar sign to place math inside a `lateximage`, or use MathJax:

`$$`

```
\begingroup
\catcode`\$=\active%
\protected\gdef${\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

Used by chemformula to escape single-dollar math:

```
\protected\gdef\LWR@newsingledollar{\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

`\LWR@doubledollar` Redefine the double dollar sign to place math inside a `lateximage`, or use MathJax:

```
\protected\gdef\LWR@doubledollar$#1$${%
```

If MathJax or formatting for a word processor, print the LaTeX expression:

```
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

For MathJax, print the math between `\[` and `\]`. If there is a footnote, endnote, or other kind of note ('`note`' is present), sync the note numbers.

```
    {% intentional blank line:

        \IfSubStr{\detokenize\expandafter{#1}}{\detokenize{note}}{%
```

The equation is printed to the PDF output inside HTML comment tags. This allows labels and footnotes to be accepted and processed. The `math` environment is selected here, and `\LWR@hidelatexequation` will use the original print-mode meaning of `math`.

```
            \LWR@hidelatexequation{math}{#1}%
```

```
            \InlineClass{hidden}{\LWR@syncnotenumbers}%
            \textbackslash[%
            {%
                \LWR@print@ttfamily%
                \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
            }%
            \textbackslash]
            \InlineClass{hidden}{\LWR@syncnotenames}%
        }{%
            \textbackslash[%
            {%
                \LWR@print@ttfamily%
                \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
            }%
            \textbackslash]
        }%

    }% mathjax
```

For SVG, print the math inside a `lateximage`, with an `<alt>` tag of the LaTeX code:

```
    {% not mathjax
        \begin{BlockClass}{displaymath}%
        \LWR@newautoidanchor%
        \booltrue{LWR@indisplaymathimage}%
        \begin{lateximage}%
        [%
            \textbackslash{[} % extra space
            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
            \textbackslash{]}%
        ]%
        *% do not add open/closing braces
        (math)% ARIA
```

Support for xfakebold:

```
        \LWR@applyxfakebold%

        \LWR@origdollar\LWR@origdollar#1\LWR@origdollar\LWR@origdollar%
        \end{lateximage}%
        \end{BlockClass}%
    }% not mathjax
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}%
```

`\LWR@singledollar` {⟨*math expression*⟩}

```
\protected\gdef\LWR@singledollar#1${%
    \LWR@traceinfo{LWR@singledollar}%
    \ifbool{mathjax}{%
        \LWR@subsingledollar*%
        {% alt tag
            \textbackslash( %
            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
```

```
            \textbackslash)%
        }%
        {singledollar}% add'l hashing
        {#1}% contents
    }{% not mathjax
        \LWR@subsingledollar*%
        {% alt tag
            \textbackslash( %
            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
            \textbackslash)%
        }%
        {singledollar}% add'l hashing
        {\LWR@origensuredmath{#1}}% contents
    }% not mathjax
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

`\(` Redefine to the above dollar macros.
`\[`

```
\AtBeginDocument{
    \protected\gdef\(#1\){$#1$}
    \protected\gdef\[#1\]{$$#1$$}
}

\endgroup% active $
```

```
\AtBeginDocument{
\LetLtxMacro\LWR@openbracketnormal\[
\LetLtxMacro\LWR@closebracketnormal\]
}
```

`\@ensuredmath` {⟨*expression*⟩}

If MathJax, a `lateximage` is used, since `\ensuremath` is often used for complex TEX expressions which MathJax may not render. If SVG math, a hashed file is used with a simple `alt` tag, but additional hashing provided by the contents.

```
\LetLtxMacro\LWR@origensuredmath\@ensuredmath

\renewcommand{\@ensuredmath}[1]{%
    \ifbool{mathjax}{%
        \LWR@subsingledollar*{\AltTextOpen\MathImageAltText\AltTextClose}%
        {%
          \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
        }%
        {%
            \relax%
            \LWR@origensuredmath{#1}%
        }%
    }{% SVG math
```

If already inside a `lateximage` in math mode, continue as-is.

```
        \ifmmode%
```

```
            \LWR@origensuredmath{#1}%
        \else%
```

Create an inline math `lateximage` with a simple `alt` tag and additional hashing according to the contents.

```
            \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
                {\LWR@origensuredmath{#1}}%
                {%
                    \LWR@subsingledollar*%
                        {\AltTextOpen\MathImageAltText\AltTextClose}%
                        {%
                            \protect\LWR@HTMLsanitizedetokenized{%
                                \detokenize\expandafter{#1}%
                            }%
                        }%
                        {\LWR@origensuredmath{#1}}%
                }%
        \fi%
    }%
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

Remember then remove the old `math` and `displaymath` environments:

```
\let\LWR@orig@math\math
\let\LWR@orig@endmath\endmath

\let\LWR@orig@displaymath\displaymath
\let\LWR@orig@enddisplaymath\enddisplaymath

\let\math\relax
\let\endmath\relax

\let\displaymath\relax
\let\enddisplaymath\relax
```

`math` (*env.*) Set math mode then typeset the body of what was between the begin/end. See the environ package for `\BODY`.

```
\NewEnviron{math}{\expandafter\(\BODY\)}
```

`LWR@displaymathnormal` (*env.*) Set math mode then typeset the body of what was between the begin/end. See the environ package for `\BODY`.

```
\NewEnviron{LWR@displaymathnormal}{\expandafter\[\BODY\]\@ignoretrue}
```

Set the default displaymath to the normal version:

```
\LetLtxMacro\displaymath\LWR@displaymathnormal%
\LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
```

LWR@displaymathother (*env.*) A version of displaymath which can handle complicated objects, but does not supply MathJax or HTML alt tags.

```
\newenvironment{LWR@displaymathother}
{%
    \begin{BlockClass}{displaymath}%
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\MathImageAltText](math)% [alt](ARIA)
    \LWR@origdollar\LWR@origdollar%
}
{%
    \LWR@origdollar\LWR@origdollar%
    \end{lateximage}%
    \end{BlockClass}%
}
```

LWR@equationother (*env.*) A version of displaymath which can handle complicated objects, but does not supply MathJax or HTML alt tags.

```
\newenvironment{LWR@equationother}
{%
    \begin{BlockClass}{displaymathnumbered}%
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\MathImageAltText](math)% [alt](ARIA)
    \LWR@orig@equation%
}
{%
    \LWR@orig@endequation%
    \end{lateximage}%
    \end{BlockClass}%
}
```

83.4 MathJax support

LWR@nextequation (*Ctr*) Used to add one to compute the next equation number.

```
\newcounter{LWR@nextequation}
```

Determing how to set MathJax section and equation numbers. Adjusts for various kinds of \theequation to determine \theMathJaxsection and \theMathJaxequation.

```
\newcommand\LWR@article@theequation{\@arabic\c@equation}

\newcommand\LWR@book@theequation
  {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}


\newcommand\LWR@chapter@theequation{\thechapter.\arabic{equation}}
\newcommand\LWR@section@thequation{\thesection.\arabic{equation}}
\newcommand\LWR@subsection@thequation{\thesubsection.\arabic{equation}}

\AtBeginDocument{
    % default per article class:
    \newcommand*{\theMathJaxsubequations}{0}
    \newcommand*{\theMathJaxsection}{}
```

```
    \newcommand*{\theMathJaxequation}{\arabic{equation}}

    \ifdefstrequal{\theequation}{\LWR@article@theequation}
    {}{
    \ifdefstrequal{\theequation}{\LWR@book@theequation}{
     \renewcommand*{\theMathJaxsection}{\ifnum \c@chapter>\z@ \thechapter.\fi}
    }{
    \ifdefstrequal{\theequation}{\LWR@subsection@thequation}{
        \renewcommand*{\theMathJaxsection}{\thesubsection{}.}
    }{
    \ifdefstrequal{\theequation}{\LWR@section@thequation}{
        \renewcommand*{\theMathJaxsection}{\thesection{}.}
    }{
    \ifdefstrequal{\theequation}{\LWR@chapter@theequation}{
        \renewcommand*{\theMathJaxsection}{\thechapter{}.}
    }{% unknown format
        \PackageWarningNoLine{lwarp}
        {%
          Unknown equation tag format for \protect\theequation.\MessageBreak
            Article-style equation numbering will be used%
        }
    }}}}}
}
```

`\LWR@syncmathjax` Sets the MathJax equation format and number for the following equations.

These MathJax commands are printed inside "`\(`" and "`\)`" characters. They are printed to HTML output, not interpreted by LaTeX.

```
\newcommand*{\LWR@syncmathjax}{%
```

Tell MathJax that the next equation number is the current LaTeX equation number.

Before each equation, lwarp inserts into the HTML code:

```
\seteqnumber{subequations?}{section}{number}
```

subequations? is `0` usually, `1` if inside amsmath `subequations`.

section is a string printed as-is, or empty.

number is auto-incremented by MathJax between equations.

Place the MathJax command inside "`\(`" and "`\)`" characters, to be printed to HTML, not interpreted by LaTeX.

```
        \LWR@stoppars%
        \InlineClass{hidden}{
            \textbackslash(%
            \textbackslash{}seteqnumber%
            \{\theMathJaxsubequations\}%
            \{\theMathJaxsection\}%
            \{\theMathJaxequation\}%
            \textbackslash)%
        }
        \LWR@startpars%
}
```

`\LWR@hidelatexequation` {⟨*environment*⟩} {⟨*contents*⟩}

Creates the LaTeX version of the equation inside an HTML comment.

```
\NewDocumentCommand{\LWR@hidelatexequation}{m +m}{%
```

Stop HTML paragraph handling and open an HTML comment:

```
    \LWR@stoppars
    \LWR@htmlopencomment

```

Start the LaTeX math environment inside the HTML comment:

```
    \begingroup
    \@nameuse{LWR@orig@#1}
```

While in the math environment, restore various commands to their LaTeX meanings.

```
    \LWR@restoreorigformatting
    \booltrue{LWR@insidemathcomment}
```

Temporarily prevent underfull \hbox warnings.

```
    \hbadness=10000\relax%
```

See `\LWR@htmlmathlabel` in section 83.7.1.

Print the contents of the equation:

```
    #2
```

End the LaTeX math environment inside the HTML comment:

```
    \@nameuse{LWR@orig@end#1}
    \endgroup

```

Close the HTML comment and resume HTML paragraph handling:

```
    \LWR@htmlclosecomment
    \boolfalse{LWR@insidemathcomment}
    \LWR@startpars
}
```

`\LWR@addmathjax` {⟨*environment name*⟩} {⟨*contents*⟩}

Given the name of a math environment and its contents, create a MathJax instance. The contents are printed to HTML output, not interpreted by LaTeX.

```
\NewDocumentCommand{\LWR@addmathjax}{m +m}{%
```

```
    \LWR@origtilde\LWR@orignewline
```

Enclose the MathJax environment inside printed "`\(`" and "`\)`" characters. Print the environment name and contents, sanitizing for HTML special characters.

```
    {%
        \LWR@print@ttfamily%
        \textbackslash{}begin\{#1\}
```

The `alignat` environment takes a mandatory argument, which must be replicated here.

```
        \ifboolexpr{
            test {\ifstrequal{#1}{alignat}} or
            test {\ifstrequal{#1}{alignat*}} or
            test {\ifstrequal{#1}{alignat+}}
        }%
            {\{\arabic{LWR@maxfields@}\}}%
            {}%
```

The environment contents and `\end`:

```
        \LWR@orignewline%
        \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{#2}}%
        \LWR@orignewline%
        \textbackslash{}end\{#1\}
    }%

    \LWR@orignewline
}
```

83.5 Equation environment

Remember existing `equation` environment, after redefined by amsmath, if loaded.

```
\AtBeginDocument{
    \let\LWR@orig@equation\equation
    \let\LWR@orig@endequation\endequation
    \csletcs{LWR@orig@equation*}{equation*}
    \csletcs{LWR@orig@endequation*}{endequation*}
}
```

`\LWR@doequation` {⟨*env contents*⟩} {⟨*env name*⟩}

For SVG math output, the contents are typeset using the original `equation` inside a `latexmage`, along with an `<alt>` tag containing a detokenized copy of the LaTeX source for the math.

For MathJax output, the contents are typeset in an original `equation` environment placed inside a HTML comment, with special processing for `\labels`. The contents are also printed to the HTML output for processing by the MathJax script.

```
\newcommand*{\LWR@doequation}[2]{%

```

If `mathjax` or `FormatWP`, print the LaTeX expression:

```
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

MathJax output:

```
    {
```

Print commands to syncronize MathJax's equation number and format to the current LaTeX chapter/section and equation number:

```
        \LWR@syncmathjax%
```

Print the LaTeX math inside an HTML comment:

```
        \LWR@hidelatexequation{#2}{#1}
    }
```

SVG output: Create the `lateximage` along with an HTML `<alt>` tag having an equation number, the LaTeX equation environment commands, and the contents of the environment's `\BODY`.

```
    {% not mathjax
```

Begin the `lateximage` with an `<alt>` tag containing the math source:

```
        \ifstrequal{#2}{equation*}{%
            \begin{BlockClass}{displaymath}%
        }{%
            \begin{BlockClass}{displaymathnumbered}%
        }%
        \LWR@newautoidanchor%
        \booltrue{LWR@indisplaymathimage}%
        \begin{lateximage}[%
            \ifstrequal{#2}{equation*}{%
                \ifdefequal{\LWR@equationtag}{\theequation}{%
    %                                   no tag was given
                }{%
                    (\LWR@equationtag) % tag was given
                }%
            }{%
                (\LWR@equationtag) % automatic numbering
            }%
            \textbackslash{begin\{#2\}} % extra space
         \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{#1}} % extra space
            \textbackslash{end\{#2\}}%
        ]*(math)% alt tag, ARIA
```

Support for xfakebold:

```
        \LWR@applyxfakebold%
```

Create the actual LaTeX-formatted equation inside the `lateximage` using the contents of the environment.

```
        \@nameuse{LWR@orig@#2}%
        #1% contents collected by \collect@body
        \@nameuse{LWR@orig@end#2}%
        \end{lateximage}%
        \end{BlockClass}%
    }% not mathjax
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

After the environment, if MathJax, print the math to the HTML output for MathJax processing. If a footnote is used, sync the `footnote` counter before, then unsync after for non-equation environments, as defined next.

```
\newcommand*{\LWR@doendequation}[1]{%
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
    {%
        \IfSubStr{\detokenize\expandafter{\BODY}}{\detokenize{note}}{%
            \InlineClass{hidden}{\LWR@syncnotenumbers}%
            \LWR@addmathjax{#1}{\BODY}%
            \InlineClass{hidden}{\LWR@syncnotenames}%
        }{%
            \LWR@addmathjax{#1}{\BODY}%
        }%
    }{}%

```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

The following are used to syncronize footnote marks and related to MathJax if `*note*` is used inside the MathJax expression. The counter is read from LaTeX then defined into MathJax for use during the following equation. After the equation, the MathJax value is returned to the text from `\footnotename`. Other notes may be added by appending to `\LWR@syncnotenumbers` and `\LWR@syncnotenames`.

`\LWR@synconenotenumber` {⟨*MathJax variable*⟩} {⟨*mark*⟩}

```
\newcommand*{\LWR@synconenotenumber}[2]{%
    \textbackslash(
    \textbackslash{}def\textbackslash{}#1\{#2\}
    \textbackslash)
}
```

`\LWR@syncnotenumbers` Assignments to make.

```
\newcommand*{\LWR@syncnotenumbers}{\LWR@synconenotenumber{LWRfootnote}{\thefootnote}}
```

`\LWR@synconenotename` {⟨*MathJax variable*⟩} {⟨*text*⟩}

```
\newcommand*{\LWR@synconenotename}[2]{%
    \textbackslash(
    \textbackslash{}def\textbackslash{}#1name\{#2\}
    \textbackslash)
}
```

`\LWR@syncnotenames` Assignments to make.

```
\newcommand*{\LWR@syncnotenames}{\LWR@synconenotename{LWRfootnote}{\footnotename}}
```

Remove existing `equation` environment:

```
\AtBeginDocument{
    \let\equation\relax
    \let\endequation\relax
    \csletcs{equation*}{relax}
    \csletcs{endequation*}{relax}
}
```

equation (*env.*) The new equation environment is created with \NewEnviron (from the environ package), which stores the contents of its environment in a macro called \BODY.

```
\AtBeginDocument{
    \NewEnviron{equation}%
        {\LWR@doequation{\BODY}{equation}}%
        [\LWR@doendequation{equation}]

    \LetLtxMacro\LWR@equationnormal\equation
    \LetLtxMacro\endLWR@equationnormal\endequation
}% AtBeginDocument
```

equation* (*env.*)

```
\AtBeginDocument{
    \NewEnviron{equation*}%
        {\LWR@doequation{\BODY}{equation*}}%
        [\LWR@doendequation{equation*}]

    \csletcs{LWR@equationnormalstar}{equation*}
    \csletcs{LWR@endequationnormalstar}{endequation*}
}% AtBeginDocument
```

Remember the "less" version of equation, which uses MathJaxand alt tags, but does not support complicated contents such as some Ti*k*Z expressions.

```
\AtBeginDocument{
    \LetLtxMacro\LWR@equationless\equation
    \LetLtxMacro\endLWR@equationless\endequation
    \csletcs{LWR@equationlessstar}{equation*}
    \csletcs{LWR@endequationlessstar}{endequation*}
}
```

83.6 \displaymathnormal and \displaymathother

\displaymathnormal By default, or when selecting \displaymathnormal, MathJax math display environments print their contents as text into HTML for MathJax to interpret, and SVG display math environments render their contents as SVG images and use their contents as the alt tag of HTML output. To do so, the contents are loaded into a macro for reuse. In some cases, such as complicated Ti*k*Z pictures, compilation will fail.

\displaymathother MathJax unsupported complicated alt tag When selecting \displaymathother, it is assumed that the contents are more complicated than "pure" math. An example is an elaborate Ti*k*Z picture, which will not render in MathJax and will not make sense as an HTML alt tag. In this mode, MathJax is turned off, math display environments become SVG images, even if MathJax is selected, and the HTML alt tags become simple messages. The contents are internally processed as an environment instead of a macro argument, so complicated objects such as Ti*k*Z pictures are more likely to compile successfully.

\displaymathnormal simple math objects

Use when display math environments have simple math which is to sent to MathJax or included in HTML `alt` tags.

```
\newcommand*{\displaymathnormal}{%
    \ifbool{LWR@origmathjax}{\booltrue{mathjax}}{\boolfalse{mathjax}}%
    \LetLtxMacro\[\LWR@openbracketnormal%
    \LetLtxMacro\]\LWR@closebracketnormal%
    \LetLtxMacro\displaymath\LWR@displaymathnormal%
    \LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
    \LetLtxMacro\equation\LWR@equationnormal%
    \LetLtxMacro\endequation\endLWR@equationnormal%
    \csletcs{equation*}{LWR@equationnormalstar}%
    \csletcs{endequation*}{LWR@endequationnormalstar}%
}
```

\displaymathother complicated math objects

Use when display math environments have complicated objects which will not work with MathJax or should not be included in HTML `alt` tags. Complicated contents are more likely to compile correctly.

```
\newcommand*{\displaymathother}{%
    \boolfalse{mathjax}%
    \LetLtxMacro\displaymath\LWR@displaymathother%
    \LetLtxMacro\enddisplaymath\endLWR@displaymathother%
    \LetLtxMacro\[\LWR@displaymathother%
    \LetLtxMacro\]\endLWR@displaymathother%
    \LetLtxMacro\equation\LWR@equationother%
    \LetLtxMacro\endequation\endLWR@equationother%
    \csletcs{equation*}{displaymath}%
    \csletcs{endequation*}{enddisplaymath}%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

Print-mode versions:

```
\newcommand*{\displaymathnormal}{}
\newcommand*{\displaymathother}{}
\newcommand*{\theMathJaxsubequations}{0}
\newcommand*{\theMathJaxsection}{}
\newcommand*{\theMathJaxequation}{\arabic{equation}}

\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

83.7 AMS Math environments

83.7.1 Support macros

LWR@amsmultline (*bool*) True if processing a multline environment.

To compensate for multline-spefific code, LWR@amsmultline is used to add extra horizontal space in \LWR@htmlmathlabel if is used in an amsmath environment which is not a multline environment and not an equation.

```
\newbool{LWR@amsmultline}
\boolfalse{LWR@amsmultline}
```

\LWR@beginhideamsmath Starts hiding LaTeX math inside an HTML comment.

```
\newcommand*{\LWR@beginhideamsmath}{
    \LWR@stoppars
    \LWR@origtilde\LWR@orignewline
    \LWR@htmlopencomment

    \begingroup
    \LWR@restoreorigformatting
```

Temporarily prevent underfull \hbox warnings.

```
    \hbadness=10000\relax%

    \booltrue{LWR@insidemathcomment}
}
```

\LWR@endhideamsmath Ends hiding LaTeX math inside an HTML comment.

```
\newcommand*{\LWR@endhideamsmath}{
    \endgroup

    \LWR@htmlclosecomment
    \boolfalse{LWR@insidemathcomment}
    \LWR@orignewline
    \LWR@startpars
}
```

83.7.2 Environment patches

The amsmath environments already collect their contents in \@envbody for further processing. eqnarray is not an $\mathcal{AMS}$ package, and thus requires special handling.

For SVG math: Each envrionment is encapsulated inside a lateximage environment, along with a special optional argument of \LWR@amsmathbody or \LWR@amsmathbodynumbered telling lateximage to use as the HTML <alt> tag the environment's contents which were automatically captured by the $\mathcal{AMS}$ environment.

For MathJax: Each environment is synched with LaTeX's equation numbers, typeset with LaTeX inside an HTML comment, then printed to HTML output for MathJax to process.

eqnarray (*env.*) This environmnet is not an $\mathcal{AMS}$ environment and thus its body is not automatically captured, so the environ package is used to capture the environment into \BODY.

```
\let\LWR@origeqnarray\eqnarray
\let\LWR@origendeqnarray\endeqnarray
```

To remember whether the starred environment was used, and thus whether to number the equations:

```
\newbool{LWR@numbereqnarray}
\booltrue{LWR@numbereqnarray}
```

Common code used by `eqnarray` and `Beqnarray` (from fancybox):

```
\newcommand{\LWR@eqnarrayfactor}{%
```

If `mathjax` or `FormatWP`, print the LaTeX expression:

```
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
    {%
```

If MathJax, the environment contents (the `\BODY`) are executed in a HTML comment to trigger the correct equation number increment (if not starred), then are included verbatim in the output for MathJax to interpret:

```
        \LWR@syncmathjax%
        \boolfalse{LWR@amsmultline}%
        \ifbool{LWR@numbereqnarray}%
        {%
```

If numbering the equations, execute a copy inside an HTML comment block:

```
            \LWR@beginhideamsmath%
            \LWR@origeqnarray%
            \BODY%
            \LWR@origendeqnarray%
            \LWR@endhideamsmath%
```

Then print the (sanitized) contents to the output for MathJax to interpret:

```
            \LWR@addmathjax{eqnarray}{\BODY}%
        }%
        {% not LWR@numbereqnarray
```

If not numbering equations, just create the contents for MathJax:

```
            \LWR@addmathjax{eqnarray*}{\BODY}%
        }% LWR@numbereqnarray
    }% mathjax
    {% not mathjax
        \ifbool{LWR@numbereqnarray}%
        {%
```

For numbered SVG equations, first create a `lateximage` with an `alt` attribute containing sanitized copy of the source code:

```
            \begin{BlockClass}{displaymathnumbered}%
            \LWR@newautoidanchor%
            \booltrue{LWR@indisplaymathimage}%
        \begin{lateximage}[(\LWR@startingequationtag\textendash\LWR@equationtag)%
                \LWR@addmathjax{eqnarray}{\BODY}]*(math)%
```

Support for xfakebold:

```
            \LWR@applyxfakebold%
```

Create the image contents using an actual eqnarray:

```
            \LWR@origeqnarray%
            \BODY%
            \LWR@origendeqnarray%
            \end{lateximage}%
            \end{BlockClass}%
        }%
        {% not LWR@numbereqnarray
```

If not numbered, do the same, but an extra \nonumber seems to be required:

```
            \begin{BlockClass}{displaymath}%
            \LWR@newautoidanchor%
            \booltrue{LWR@indisplaymathimage}%
            \begin{lateximage}[\LWR@addmathjax{eqnarray*}{\BODY}]*(math)%
```

Support for xfakebold:

```
            \LWR@applyxfakebold%

            \def\@eqncr{\nonumber\@seqncr}
            \csuse{LWR@origeqnarray}%
            \BODY%
            \nonumber\csuse{LWR@origendeqnarray}%
            \end{lateximage}%
            \end{BlockClass}%
        }% LWR@numbereqnarray
    }% not mathjax
```

Default to number equations in the future:

```
    \booltrue{LWR@numbereqnarray}%
```

Clear the single-use alt text:

```
    \gdef\LWR@ThisAltText{}%
}
```

eqnarray itself is made with a blank line before and after to force it to be on its own line:

```
\RenewEnviron{eqnarray}
{%

\LWR@eqnarrayfactor

}
```

The starred version is patched to turn off the numbering:

```
\csgpreto{eqnarray*}{\boolfalse{LWR@numbereqnarray}}

\end{warpHTML}
```

84 Lateximages

84.1 Description

lateximage (*env.*) A lateximage is a piece of the document which is typeset in LaTeX then included in the HTML output as an image. This is used for math if SVG math is chosen, and also for the picture, tikzpicture, and other environments.

Before typesetting the lateximage a large number of formatting, graphics, and symbols-related macros are temporarily restored to their print-mode meaning by \LWR@restoreorigformatting. (See section 81.)

A lateximage is typeset on its own PDF page inside an HTML comment which starts on the preceeding page and ends on following page, and instructions are written to lateximage.txt for *lwarpmk* to extract the lateximage from the page of the PDF file then generate an accompanying .svg file image file. Meanwhile, instructions to show this image are placed into the HTML file after the comment.

An HTML is created to hold both the HTML comment, which will have the *pdftotext* conversion, and also the link to the final .svg image.

A LaTeX label is used to remember which PDF page has the image. A label is used because footnotes, endnotes, and pagenotes may cause the image to appear at a later time. The label is declared along with the image, and so it correctly remembers where the image finally ended up.

HTML alt tag The HTML alt tag is set to the LaTeX source for SVG math, some chemistry expressions, and perhaps some other expressions which make sense for text copy/paste. In some other cases, the alt tag is set according to the package name.

When creating an SVG math image, its HTML alt tag may be set to the math expression, which may be hashed for image reuse. In the case of \ensuremath or after \inlinemathother, where the contents require a unique image for each instance of the same expression, the alt tag is set to \MathImageAltText, along with \AltTextOpen and \AltTextClose, and the image is not reused.

This alt expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "math image", and it may be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following SVG math images.

For many packages, the output is placed inside a lateximage with an HTML alt tag set to the package name followed by \PackageDiagramAltText. For example:

```
(-xy- diagram)
```

This expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "diagram", and may it be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following package diagrams.

SVG image font size For the lateximage environment, the size of the math and text used in the SVG image may be adjusted by setting \LateximageFontSizeName to a font size name — *without the backslash*, which defaults to:

```
\renewcommand{\LateximageFontSizeName}{normalsize}
```

For inline SVG math, font size is instead controlled by `\LateximageFontScale`, which defaults to:

```
\newcommand*{\LateximageFontScale}{.75}
```

84.2 Support counters and macros

for HTML output:

```
\begin{warpHTML}
```

`LWR@lateximagenumber` (*Ctr*) Sequence the images.

```
\newcounter{LWR@lateximagenumber}
\setcounter{LWR@lateximagenumber}{0}
```

`LWR@lateximagedepth` (*Ctr*) Do not create `\lateximage` inside of `\lateximage`.

```
\newcounter{LWR@lateximagedepth}
\setcounter{LWR@lateximagedepth}{0}
```

A few utility macros to write special characters:

```
\edef\LWR@hashmark{\string#} % for use in \write
\edef\LWR@percent{\@percentchar} % for use in \write
```

`LWR@LIpage` (*Ctr*) Used to reference the PDF page number of a lateximage to be written into `<project>-images.txt`.

```
\newcounter{LWR@LIpage}
```

```
\end{warpHTML}
```

84.3 Font size

for HTML & PRINT:

```
\begin{warpall}
```

`\LateximageFontSizeName` Declares how large to write text in `\lateximages`. The `.svg` file text size should blend well with the surrounding HTML text size.

⚠ no backslash *Do not include the leading backslash in the name.*

```
\newcommand*{\LateximageFontSizeName}{normalsize}
```

`\LateximageFontScale` Declares how large to scale inline SVG math images. The `.svg` file text size should blend well with the surrounding HTML text size. The default is 1, but it may be redefined as needed depending on the HTML font.

```
\newcommand*{\LateximageFontScale}{1}
```

```
\end{warpall}
```

84.4 Equation numbers

for HTML output:

```
\begin{warpHTML}
```

LWR@startingequation (*Ctr*) For use with lateximage and multi-line numbered equations. Remembers the next equation number so that it may be printed in the alt tag.

```
\newcounter{LWR@startingequation}

\@ifundefined{chapter}
{
\renewcommand{\theLWR@startingequation}{%
    \arabic{LWR@startingequation}%
}
}
{% chapter defined
\renewcommand{\theLWR@startingequation}{%
    \ifnumcomp{\value{chapter}}{>}{0}{\arabic{chapter}.}{}%
    \arabic{LWR@startingequation}%
}
}
```

LWR@isstartingequation (*bool*) True for the first equation tag, false for later tags in the same environment.

```
\newbool{LWR@isstartingequation}
```

\LWR@startingequationtag Prints the starting equation number or tag.

```
\let\LWR@startingequationtag\theLWR@startingequation
```

\LWR@equationtag Prints the ending equation number or tag.

This is reset by lateximage, may be temporarily overwritten by \tag calling \LWR@remembertag.

```
\newcommand*{\LWR@equationtag}{}
```

Only if SVG math, patch \tag after packages have loaded, in case someone else modified \tag.

```
\AtBeginDocument{

\ifbool{mathjax}{}{% not mathjax
```

\LWR@remembertag {⟨*tag*⟩}

For use inside the math environments while using SVG math. Sets \theLWR@startingequation and \theequation to the given tag.

```
\NewDocumentCommand{\LWR@remembertag}{m}{%
    \ifbool{LWR@isstartingequation}%
    {%
        \global\boolfalse{LWR@isstartingequation}%
        \xdef\LWR@startingequationtag{#1}%
    }{}%
```

```
    \xdef\LWR@equationtag{#1}%
}%

}% not mathjax
}% AtBeginDocument
```

84.5 HTML alt tags

`\LWR@amsmathbody` {⟨*envname*⟩} For use inside the optional argument to a `lateximage` to add the contents of a AMS math environment to the `<alt>` tag.

```
\newcommand*{\LWR@amsmathbody}[1]
{%
    \textbackslash\{begin\}\{#1\} % extra space
    \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{\the\@envbody}}%
    \textbackslash\{end\}\{#1\}%
}
```

`\LWR@amsmathbodynumbered` {⟨*envname*⟩} For use inside the optional argument to a `lateximage` to add the contents of a AMS math environment to the `alt` tag, prefixed by the equation numbers.

```
\newcommand*{\LWR@amsmathbodynumbered}[1]
{%
    \ifnumcomp{\value{LWR@startingequation}}{=}{\value{equation}}%
        {(\LWR@equationtag)}%
       {(\LWR@startingequationtag\textendash\LWR@equationtag)} % extra space
    \LWR@amsmathbody{#1} % extra space
}
```

84.6 lateximage environment

`\LWR@lateximage@oneimageb` {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: css style*⟩} {⟨*4: aria role*⟩} Creates the image for the `lateximage`.

```
\newcommand{\LWR@lateximage@oneimageb}[4]{%
    \LWR@subinlineimage{#1}{lateximage}%
    {%
        \LWR@print@mbox{%
            \LWR@ImagesDirectory\OSPathSymbol%
            #2%
        }%
    }{svg}{#3}{#4}%
}
```

`\LWR@lateximage@oneimage` {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: css style*⟩} {⟨*4: delimit?*⟩} {⟨*5: aria role*⟩}

Creates an image for the `lateximage`, whose `alt` text depends on the circumstances.

```
\newcommand{\LWR@lateximage@oneimage}[5]{%
    \LWR@traceinfo{LWR@lateximage@oneimage !#1!#2!#3!#4!#5!}%
    \ifdefvoid{\LWR@ThisAltText}{%
```

```
        \IfBooleanTF{#4}{%
            \LWR@lateximage@oneimageb{#1}{#2}{#3}{#5}%
        }{%
            \LWR@lateximage@oneimageb%
                {\AltTextOpen#1\AltTextClose}%
                {#2}{#3}{#5}%
        }%
    }{%
        \LWR@lateximage@oneimageb%
            {\AltTextOpen\LWR@ThisAltText\AltTextClose}%
            {#2}{#3}{#5}%
    }%
}
```

`lateximage` (*env.*) * [⟨*2:* <alt> *tag*⟩] * [⟨*4: add'l hashing*⟩] [⟨*5: css style*⟩] (⟨*6: aria role*⟩)

Typesets the contents and then renders the result as an SVG file. Star #1 causes the image to be hashed for reuse. Star #3 causes the `alt` tag to not include `\AltTextOpen` and `\AltTextClose`, for use with math expressions.

The optional <alt> tag is included in the HTML code for use with copy/paste.

image filename hashing

If starred, a hashed filename is used.If so, the hash is based on the `alt` tag and also the additional hashing argument.

This may be used to provide an expression with a simple `alt` tag but also enough additional information to provide a unique hash.

An example is when the expression is a complicated TEX expression, which would not copy/paste well. A simplified tag may be used, while the complicated expression is used in the additional hashing argument to ensure a unique image.

Another example is when the expression is simple, but the image depends on options. These options may be decoded into text form and included in the additional hashing argument in order to make the hash unique according to the set of options, even if the simple `alt` tag is still the same.

`*_html.aux` (*file*) A new label is placed into the file `*_html.aux`:

```
\newlabel{LWRlateximage-<BaseJobname>-<number>}{{<x>}{<y>}}
```

This is used to find the image in the PDF file, according to its name.

`*-images.txt` (*file*) A list of images to generate is created in <jobname>-images.txt. Each line has three pipe-delimited fields, containing the PDF page number from <jobname>_html.pdf, where the image is located, a boolean indicating whether the image is hashed, and the filename of the image. The last line has "end" in each field, and is used to detect an incomplete compile.

```
\catcode`\$=\active%

\NewDocumentEnvironment{lateximage}{s O{\ImageAltText} s O{} O{} D(){}}%
{%
\LWR@traceinfo{lateximage !#1!#2!#3!#4!#5!#6!}%
\LWR@traceinfo{lateximage: starting on \jobname.pdf page \arabic{page}}%
\LWR@traceinfo{lateximage: entering depth is \arabic{LWR@lateximagedepth}}%
```

Nested lateximages remain one large lateximage:

```
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
```

If nesting inside an already-existing lateximage, simply record one more level. $\mathcal{AMS}$ packages redefine \addtocounter to do nothing if inside a \text, so lower-level TEX macros are used for tracking nested lateximages.

```
{%
%     \addtocounter{LWR@lateximagedepth}{1}%
    \global\advance\c@LWR@lateximagedepth 1\relax% Due to AmS \text macro.
}%
```

Otherwise, this is the outer-most lateximage:

```
{% start of outer-most lateximage
```

Remember the next equation number to be allocated, in case it must be printed in a multi-equation environment:

```
    \LWR@traceinfo{lateximage: starting outer-most lateximage}%
    \setcounter{LWR@startingequation}{\value{equation}}%
    \addtocounter{LWR@startingequation}{1}%
    \booltrue{LWR@isstartingequation}%
    \let\LWR@startingequationtag\theLWR@startingequation%
```

The default equation tag, unless overwritten by \tag:

```
    \let\LWR@equationtag\theequation%
```

Starting a new lateximage:

```
    \addtocounter{LWR@lateximagenumber}{1}%
   \LWR@traceinfo{lateximage: LWR@lateximagenumber is \arabic{LWR@lateximagenumber}}%
```

While inside a lateximage, locally do not use mathjax:

```
    \boolfalse{mathjax}%
```

Be sure that are doing a paragraph:

```
    \LWR@ensuredoingapar%
```

Inside the lateximage, temporarily prevent underfull \hbox warnings.

```
    \hbadness=10000\relax%
```

Next file:

```
    \addtocounter{LWR@externalfilecnt}{1}%
   \LWR@traceinfo{lateximage: LWR@externalfilecnt is \arabic{LWR@externalfilecnt}}%
```

Figure out what the next page number will be. \setcounterpageref assigns LWR@LIpage to the page number for the reference LWRlateximage-BaseJobname-XXX:

```
    \setcounterpageref{LWR@LIpage}{%
        LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
```

```
    }%
    \LWR@traceinfo{lateximage: LWR@LIpage is \arabic{LWR@LIpage}}%
```

Create an HTML span which will hold the comment which contains the *pdftotext* translation of the image's page, and also will hold the link to the .svg file:

```
    \LWR@htmltag{span\LWR@indentHTML%
        id=\textquotedbl{}%
            lateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
            \textquotedbl\LWR@indentHTML
        class=\textquotedbl{}lateximagesource\textquotedbl\LWR@orignewline
    }%
```

Write instructions to the <ImagesDirectory>.txt file:

```
    \LWR@traceinfo{lateximage: about to write to \BaseJobname-images.txt}%
    \IfBooleanTF{#1}% starred
    {% hash
```

Compute and save the hashed file name for later use:

```
        \ifdefvoid{\LWR@ThisAltText}{%
            \IfBooleanTF{#3}{%
                \edef\LWR@hashedname{%
                    \LWR@mdfive{\detokenize\expandafter{#2}-!-#4}%
                }%
            }{%
                \edef\LWR@hashedname{%
              \LWR@mdfive{\detokenize\expandafter{\AltTextOpen#2\AltTextClose}-!-#4}%
                }%
            }%
        }{%
            \edef\LWR@hashedname{%
           \LWR@mdfive{\detokenize\expandafter{\AltTextOpen\LWR@ThisAltText\AltTextClose}-!-#4}%
            }%
        }%
        \LWR@traceinfo{lateximage: hash is \LWR@hashedname}%
```

Write the page, hashing, and hashed name:

```
        \immediate\write\LWR@lateximagesfile{%
            |\arabic{LWR@LIpage}|true|\LWR@hashedname|%
        }%
    }% hash
    {% no hash
```

No hash, so write the page, no hashing, and the image number:

```
        \LWR@traceinfo{lateximage: hash false}%
        \immediate\write\LWR@lateximagesfile{%
        |\arabic{LWR@LIpage}|false|\LWR@ImagesName\arabic{LWR@externalfilecnt}|%
        }%
    }% no hash
```

Place an open comment tag. This will hide any traces of the lateximage PDF page which were picked up by *pdftotext*.

```
    \LWR@traceinfo{lateximage: about to create open comment}%
    \LWR@htmlopencomment%
```

One level deeper. At this outer-most lateximage, it is known that this is not being used inside an $\mathcal{AMS}$ \text, since the outer-most level will never be in math mode.

```
12546      \addtocounter{LWR@lateximagedepth}{1}%
```

Start the new PDF page:

```
12547     \LWR@traceinfo{lateximage: about to create a new page}%
12548     \LWR@maybe@orignewpage%
```

If the current page is larger, typeset the image in a "standard" width page and font size:

```
12549     \LWR@traceinfo{lateximage: about to create minipage}%
12550     \setcounter{LWR@mpfootnote@store}{\value{mpfootnote}}
12551     \ifdimless{\linewidth}{6in}{%
12552         \LWR@print@minipage{\linewidth}%
12553     }{%
12554         \LWR@print@minipage{6in}%
12555     }%

12556     \ifnumgreater{\value{LWR@minipage@depth}}{0}%
12557         {\setcounter{mpfootnote}{\value{LWR@mpfootnote@store}}}%
12558         {}%
12559     \@nameuse{LWR@print@\LateximageFontSizeName}%
```

Temporarily restore formatting to its PDF definitions: Do not produce HTML tags for \hspace, etc. inside a lateximage.

```
12560     \LWR@traceinfo{lateximage: about to temporarily restore formatting}%
12561     \LWR@restoreorigformatting%
```

If not inside a minipage, use full-page footnotes instead of minipage footnotes. These become HTML footnotes.

```
12562     \ifnumgreater{\value{LWR@minipage@depth}}{0}%
12563         {}%
12564         {%
12565             \def\@mpfn{footnote}%
12566             \def\thempfn{\thefootnote}%
12567             \LetLtxMacro\@footnotetext\LWR@footnotetext%
12568         }%
```

Create the LWRlateximage<number> label:

```
12569     \LWR@traceinfo{lateximage: about to create label}%
12570    \LWR@orig@label{LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}}%
12571     \LWR@traceinfo{lateximage: finished creating the label}%
```

Adjust the rule color to match HTML:

```
12572     \ifdefvoid{\LWR@ruleHTMLcolor}{}{%
12573         \LWR@print@arrayrulecolor[HTML]{\LWR@ruleHTMLcolor}%
12574     }%
```

Enable print-mode math functions:

```
    \LetLtxMacro$\LWR@origdollar%
    \catcode`\$=3% math shift
    \LetLtxMacro\(\LWR@origopenparen%
    \LetLtxMacro\)\LWR@origcloseparen%
```

Only enable print-mode display math if are not already inside display math:

```
    \ifbool{LWR@indisplaymathimage}{}{% not in display math
        \LetLtxMacro\[\LWR@origopenbracket%
        \LetLtxMacro\]\LWR@origclosebracket%
        \let\equation\LWR@orig@equation%
        \let\endequation\LWR@orig@endequation%
        \csletcs{equation*}{LWR@orig@equation*}%
        \csletcs{endequation*}{LWR@orig@endequation*}%
    }% not in display math
```

For chemformula:

```
    \LetLtxMacro\LWR@newsingledollar$%
    \LetLtxMacro\LWR@newsingledollar$% syntax highlighting
```

```
}% end of outer-most lateximage
\LWR@traceinfo{lateximage: finished start of environment}%
}% end of \begin{lateximage}
```

\endlateximage When the lateximage environment closes:

```
{% start of \end{lateximage}
\LWR@traceinfo{lateximage: starting end of lateximage}%
```

Nested more than one deep?

```
\LWR@traceinfo{lateximage: internal depth was \arabic{LWR@lateximagedepth}}%
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{1}%
```

If nesting inside an already existing lateximage, simply record one less level. Uses a lower-level TEX macro due to $\mathcal{AMS}$ \text change of \addtocounter.

```
{%
    \LWR@traceinfo{lateximage: unnesting}%
    \global\advance\c@LWR@lateximagedepth -1\relax%
}%
```

If this is the outer-most lateximage:

```
{% end of outer-most lateximage
```

Finish the lateximage minipage and start a new PDF page:

```
\LWR@traceinfo{lateximage: ending outer-most lateximage}%
    \endLWR@print@minipage%
    \LWR@maybe@orignewpage%
```

Close the HTML comment which encapsulated any traces of the lateximage picked up by *pdftotext*:

```
    \LWR@print@vspace*{.5\baselineskip}%
```

```
    \LWR@htmlclosecomment%
    \LWR@traceinfo{lateximage: The page after the image is \arabic{page}}%
```

Create a link to the lateximage, allowing its natural height:

```
    \LWR@traceinfo{about to LWR@lateximage@oneimage !#2!}%
    \IfBooleanTF{#1}% starred
    {% hash
        \LWR@lateximage@oneimage{#2}{\LWR@hashedname}{#5}{#3}{#6}%
    }% hash
    {% no hash
     \LWR@lateximage@oneimage{#2}{\LWR@ImagesName\theLWR@externalfilecnt}{#5}{#3}{#6}%
    }% no hash
```

Be sure that are doing a paragraph:

```
    \LWR@ensuredoingapar%
```

Close the HTML span which has the *pdftotext* comment and also the link to the `.svg` image:

```
    \LWR@htmltag{/span}%
    \ifbool{HTMLDebugComments}{%
        \LWR@htmlcomment{End of lateximage}%
    }{}%
```

Undo one lateximage level. This is not inside an $\mathcal{AMS}$ `\text`, so regular `\addtocounter` may be used here.

```
    \addtocounter{LWR@lateximagedepth}{-1}%
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}% end of outer-most lateximage
\LWR@traceinfo{lateximage: exiting depth is \arabic{LWR@lateximagedepth}}%
\LWR@traceinfo{lateximage: done}%
}%
\catcode`\$=3% math shift
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

`lateximage` (*env.*) * [⟨<alt> *tag*⟩] * [⟨*add'l hashing*⟩] [⟨*css style*⟩]

Ignored in print mode.

```
\NewDocumentEnvironment{lateximage}{s o s o o d()}
    {}{}

\end{warpprint}
```

85 `center`, `flushleft`, `flushright`

for HTML output:

```
\begin{warpHTML}
```

center (*env.*) Replace center functionality with css tags. In a , these macros are nullified, but extra % are used to remove spurrious spaces here as well.

```
\newenvironment*{LWR@HTML@center}
{%
    \LWR@forcenewpage%
    \ifbool{FormatWP}%
        {\BlockClass[\LWR@print@mbox{text-align:center}]{center}}%
        {\BlockClass{center}}%
}
{\endBlockClass}

\LWR@formattedenv{center}
```

flushright (*env.*)

```
\newenvironment*{LWR@HTML@flushright}
{%
    \LWR@forcenewpage%
    \ifbool{FormatWP}%
        {\BlockClass[\LWR@print@mbox{text-align:right}]{flushright}}%
        {\BlockClass{flushright}}%
}
{\endBlockClass}

\LWR@formattedenv{flushright}
```

flushleft (*env.*)

```
\newenvironment*{LWR@HTML@flushleft}
{%
    \LWR@forcenewpage%
    \ifbool{FormatWP}%
        {\BlockClass[\LWR@print@mbox{text-align:left}]{flushleft}}%
        {\BlockClass{flushleft}}%
}
{\endBlockClass}

\LWR@formattedenv{flushleft}
```

\centering, \raggedleft, and \raggedright usually have no effect on the HTML output, but they may be used to compare with the next token to identify their use at the start of a float. See \LWR@floatalignment.

\centering

```
\newcommand*{\LWR@HTML@centering}{%
    \ifbool{HTMLDebugComments}{%
        \LWR@htmlcomment{centering}%
    }{}%
}
\LWR@formatted{centering}
```

\raggedleft

```
\newcommand*{\LWR@HTML@raggedleft}{%
```

```
    \ifbool{HTMLDebugComments}{%
        \LWR@htmlcomment{raggedleft}%
    }{}%
}
\LWR@formatted{raggedleft}
```

\raggedright

```
\newcommand*{\LWR@HTML@raggedright}{%
    \ifbool{HTMLDebugComments}{%
        \LWR@htmlcomment{raggedright}%
    }{}%
}
\LWR@formatted{raggedright}
```

\leftline {⟨*text*⟩}

```
\renewcommand{\leftline}[1]{\begin{flushleft}#1\end{flushleft}}
```

\centerline {⟨*text*⟩}

```
\renewcommand{\centerline}[1]{\begin{center}#1\end{center}}
```

\rightline {⟨*text*⟩}

```
\renewcommand{\rightline}[1]{\begin{flushright}#1\end{flushright}}
```

```
\end{warpHTML}
```

86 Preloaded packages

for HTML output:

```
\begin{warpHTML}
```

If the given package was loaded before or by lwarp, load the lwarp version as well.

\LWR@PreloadedPackage {⟨*packagename*⟩}

```
\newcommand*{\LWR@PreloadedPackage}[1]{%
    \IfPackageLoadedTF{#1}%
        {%
            \AtBeginDocument{
                \LWR@origRequirePackage{lwarp-#1}%
            }
        }%
        {}%
}
```

If inputtrc was loaded before lwarp, as is usually done, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{inputtrc}
```

If textcomp was loaded before lwarp, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{textcomp}
```

If xunicode was loaded before lwarp, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{xunicode}
```

If graphics or graphicx were loaded before lwarp, perhaps by xunicode, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{graphics}
\LWR@PreloadedPackage{graphicx}
```

tagpdf-base may have been preloaded by pdfmanagement-testphase

```
\LWR@PreloadedPackage{tagpdf-base}
```

scalefnt may have been preloaded by babel

```
\LWR@PreloadedPackage{scalefnt}
```

fontaxes must be preloaded so that lwarp may patch it for HTML.

```
\LWR@PreloadedPackage{fontaxes}
```

Various font packages which may be loaded before lwarp:

```
\LWR@PreloadedPackage{cmbright}
\LWR@PreloadedPackage{fourier}
\LWR@PreloadedPackage{kpfonts}
\LWR@PreloadedPackage{kpfonts-otf}
\LWR@PreloadedPackage{libertinust1math}
\LWR@PreloadedPackage{pxfonts}
\LWR@PreloadedPackage{txfonts}
\LWR@PreloadedPackage{txgreeks}
\LWR@PreloadedPackage{newpxmath}
\LWR@PreloadedPackage{newtxmath}
\LWR@PreloadedPackage{newtxsf}
\LWR@PreloadedPackage{mathalpha}
\LWR@PreloadedPackage{unicode-math}
```

```
\LWR@PreloadedPackage{realscripts}
```

nfssext-cfr may be preloaded by cfm-lm or related font packages.

```
\LWR@PreloadedPackage{nfssext-cfr}
```

ulem may be preloaded by ctex, ctexart, and related classes.

```
\LWR@PreloadedPackage{ulem}
```

```
\LWR@PreloadedPackage{xetexko}
```

geometry is preloaded by lwarp, and perhaps by various classes.

```
\LWR@PreloadedPackage{geometry}
```

plext is preloaded by some CJK classes.

```
\LWR@PreloadedPackage{plext}
```

stfloats is preloaded by ltj* classes.

```
\LWR@PreloadedPackage{stfloats}
```

lltjext is preloaded by ltj* classes.

```
\LWR@PreloadedPackage{lltjext}
```

luatexko must be loaded before lwarp.

```
\LWR@PreloadedPackage{luatexko}
```

```
\end{warpHTML}
```

87 siunitx

`siunitx` (*Pkg*)

A few HTML unit equivalents are defined here.

siunitx is well supported by lwarp.

Limitations Some general limitations:

fractions Due to *pdftotext* limitations, fraction output is replaced by symbol output for `per-mode` and `quotient-mode`.

`\cancel` is not currently supported for siunitx v3.

Negative values are not automatically colored.

⚠ `tabular` Tabular `S` and `s` columns are rendered as simple `c` columns, although key settings will be set. If using scientific notation, `table-format`, `table-align-uncertainty`, `drop-exponent`, etc.. use `\tablenum` for each cell. This is especially required for

⚠ `drop-exponent` `drop-exponent`, without which the value will be shown incorrectly.

⚠ `table-auto-round` `table-auto-round` is ignored.

Math rendering Math may be rendered in several ways in the same document:

For math mode with SVG display: The original siunitx code is used while generating the SVG image.

For HTML text mode: lwarp uses siunitx code patched for HTML, and simplified units.

For math expressions while using MathJax: A limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. The result usually looks fine, and otherwise is enough to get the meaning across.

Custom units siunitx allows customized units:

`\DeclareSIUnit` {⟨*name*⟩} {⟨*definition*⟩}

`\DeclareSIUnit` declares a version of the unit for the print version. This is also used when the unit is printed in SVG math or a `lateximage`. It is also used for HTML if an HTML-specific version is not defined with `\HTMLDeclareSIUnit`.

```
\DeclareSIUnit\myunit{\ensuremath{\text{m}_y}}
```

`\HTMLDeclareSIUnit` {⟨*name*⟩} {⟨*definition*⟩}

⚠ v3 only! Use this after the print unit has been defined. For siunitx v3, `\HTMLDeclareSIUnit` declares a simplified version of the unit for HTML, for example if the print-mode unit uses TEX boxes or `\ensuremath`:

```
\HTMLDeclareSIUnit\myunit{\text{m}\textsubscript{\textit{y}}}
```

It is also possible to provide a custom unit for MathJax:

```
\CustomizeMathJax{\newcommand{\myunit}{\text{m}_y}}
```

Predefined units Most units work as-is with HTML. For the following units, lwarp has already set `\HTMLDeclareSIUnit`: `\celsius`, `\arcminute`, `\arcsecond`, `\elementarycharge`, `\clight`, `\bohr`, `\electronmass`, `\hartree`, `\planckbar`.

⚠ MathJax

Document modifications required for MathJax

⚠ `\sisetup`
- Place `\sisetup` in the preamble before `\begin{document}`. Changes made later may be ignored, especially with MathJax. The MathJax emulation also ignores most macro options.

⚠ complex numbers
- Complex numbers are displayed as entered, ignoring `output-complex-root`.

custom units
- Custom units may be added with `\CustomizeMathJax`. For example, from lwarp-common-mathjax-siunitx:

```
\CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
\CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
```

⚠ unit spacing
- Units work better using ~ between units instead of using periods.

⚠ `\square, \cubic`
- To square or cube compound units, enclose the following compound units in braces:

```
\cubic{\centi\meter}
```

  Single units do not require braces.

- For `\numlist`, the argument is printed as text as-is, so use space between semicolons for improved readability.

⚠ `Missing $ inserted`
- If using `parse-numbers = false`, also use `\num` or `\qty`. siunitx=siunitx>Missing $ inserted.

Also see MathJax option, section 8.7.5.

for HTML output:

```
\begin{warpHTML}
```

Options for siunitx:

```
\newrobustcmd{\LWR@siunitx@textcelsius}{\HTMLentity{deg}C}
\newrobustcmd{\LWR@siunitx@textdegree}{\HTMLentity{deg}}
\newrobustcmd{\LWR@siunitx@textprime}{\HTMLunicode{2032}}
\newrobustcmd{\LWR@siunitx@textdblprime}{\HTMLunicode{2033}}
\newrobustcmd{\LWR@siunitx@textplanckbar}{\text{\textit{\HTMLunicode{210F}}}}

\appto\LWR@restoreorigformatting{%
\renewrobustcmd{\LWR@siunitx@textcelsius}{\text{\ensuremath{^\circ}C}}%
\renewrobustcmd{\LWR@siunitx@textdegree}{\text{\ensuremath{^\circ}}}%
\renewrobustcmd{\LWR@siunitx@textprime}{\text{\ensuremath{^\prime}}}%
\renewrobustcmd{\LWR@siunitx@textdblprime}{\text{\ensuremath{^{\prime\prime}}}}%
\renewrobustcmd{\LWR@siunitx@textplanckbar}{\text{\ensuremath{\hbar}}}%
}

\end{warpHTML}
```

for PRINT output: The print version of `\HTMLDeclareSIUnit`.

```
\begin{warpprint}
\NewDocumentCommand{\HTMLDeclareSIUnit}{o +m m}{}
\end{warpprint}
```

88 Graphics print-mode modifications

88.1 General limitations

file extensions

Per table 9, image filenames may be specified either with or without an extension. If an extension is given it will be used as-is, for either print or HTML output. If no extension is given, a list of possible extensions is tried, which depends on whether print or HTML is being generated. This allows a PDF file for print and a SVG file for HTML, for example. If no extension is given, the automatic search will only return lowercase extensions, even if the filename actually has an uppercase extension, and lwarp cannot get around this problem, so image file extensions must be lowercase to be seen by the HTML browser with lwarp. For example, name the image file `image.pdf` instead of `image.PDF`, but refer to it in the source as `image`, without an extension. For images which may be used as-is with either print or HTML, such as JPG or PNG, you may use a capitalized extension if it is specified in the source, such as `image.JPG`.

⚠ case sensitive

`\includegraphics` file formats

For `\includegraphics` with `.pdf` or `.eps` files, the user must provide a `.pdf` or `.eps` image file for use in print mode, and also a `.svg`, `.png`, or `.jpg` version of the same image for use in HTML.

```
\includegraphics{filename} % print:.pdf/.eps HTML:.svg, etc.
```

For print output, lwarp will automatically choose the `.pdf` or `.eps` format if available, or some other format otherwise. For HTML, one of the other formats is used instead.

If a .pdf or .eps image is referred to with its file extension, the extension will be changed to .svg for HTML:

```
\includegraphics{filename.pdf} % uses .svg in html
\includegraphics{filename.eps} % uses .svg in html
```

pdftocairo (*Prog*)
PDF to SVG

To convert a PDF image to SVG, use the utility *pdftocairo*:

Enter ⇒ **pdftocairo -svg filename.pdf**

lwarpmk pdftosvg (*Prog*)

For a large number of images, use *lwarpmk*:

Enter ⇒ **lwarpmk pdftosvg *.pdf** **(or a list of filenames)**

lwarpmk epstopdf (*Prog*)
epstopdf (*Prog*)
epstopdf package

For EPS images converted to PDF using the package epstopdf, use

Enter ⇒ **lwarpmk pdftosvg *.PDF**

to convert to SVG images.

DVI LaTeX

When using DVI *latex*, it is necessary to convert EPS to PDF and then to SVG:

Enter ⇒ **lwarpmk epstopdf *.eps** **(or a list of filenames)**

Enter ⇒ **lwarpmk pdftosvg *.pdf** **(or a list of filenames)**

PNG and JPG

For PNG or JPGwhile using *pdflatex*, *lualatex*, or *xelatex*, the same file may be used in both print or HTML versions, and may be used with a file extension, but will also be used without the file extension if it is the only file of its base name.

GIF

GIF files may be used for HTML, but another format must also be provided for print output.

file extension priorities

If a file extension is not used, for HTML the file extension priorities are: SVG, GIF, PNG, then JPG.

duplicate files
⚠ image not displayed

A complication occurs if a file of the same name exists elsewhere in the TeX tree, such as a test image from some LaTeX package. TeX looks in the local document directory before considering the directories specified by \graphicspath, but the TeX tree is found as "local", so any file in the tree is found before the directories in \graphicspath. To use such an image, it must be copied to the document's directory to be used for HTML, and furthermore must be in the document's base directory instead of an images subdirectory.

⚠ graphics vs. graphicx

If using the older graphics syntax, use both optional arguments for \includegraphics. A single optional parameter is interpreted as the newer graphicx syntax. Note that

⚠ viewport

viewports are not supported by lwarp — the entire image will be shown.

units

For \includegraphics, avoid px and % units for width and height, or enclose them inside warpHTML environments. For font-proportional image sizes, use ex or em. For fixed-sized images, use cm, mm, in, pt, or pc. Use the keys width=.5\linewidth, or similar for \textwidth or \textheight to give fixed-sized images proportional to a 6 by 9 inch text area. Do not use the scale option, since it is not well supported by HTML browsers.

options

\includegraphics accepts width and height, origin, rotate and scale, plus new class and alt keys. (alt has recently been incorportated into graphicx itself.)

HTML class

With HTML output, \includegraphics accepts an optional class=xyz keyval com-

bination, and if this is given then the HTML output will include that class for the image. The class is ignored for print output.

HTML `alt` tags

Likewise, the `\includegraphics` `alt` key adds an HTML `alt` tag to an image, and is ignored for print output. If not assigned, each image is given an `alt` tag according to `\ImageAltText`.

⚠ `scale`

Avoid using the `\includegraphics` `scale` option. Change:

```
\includegraphics[scale=<xx>]{ . . . }
```

to:

```
\includegraphics[width=<yy>\linewidth]{ . . . }
```

`\rotatebox`

`\rotatebox` accepts the optional `origin` key.

⚠ browser support

`\rotatebox`, `\scalebox`, and `\reflectbox` depend on modern browser support. The css3 standard declares that when an object is transformed the whitespace which they occupied is preserved, unlike LaTeX, so expect some ugly results for scaling and rotating.

88.2 Print-mode modifications

for PRINT output:

For print output, accept and then discard the new `class` key:

```
\begin{warpprint}
\define@key{Gin}{class}{}
```

Print-mode additions for the overpic package. See section 460 for the HTML version.

```
\AtBeginDocument{
\IfPackageLoadedTF{overpic}{
\newcommand*{\overpicfontsize}{12}
\newcommand*{\overpicfontskip}{14}
}{}
}
\end{warpprint}
```

89 xcolor boxes

`xcolor` (*Pkg*)

A few new definitions are provided for enhanced HTML colored boxes, and `\fcolorbox` is slightly modified. Print-mode version are also provided.

Print-mode versions of new xcolor defintions. These are defined inside `warpall` because they are also used for HTML while inside a `lateximage`. They are defined `\AtBeginDocument` so that the xcolor originals may first be loaded and saved for reuse.

The framed versions are modified to allow a background color of `none`, in which case only the frame is drawn, allowing the background page color to show.

for HTML & PRINT:

```
\begin{warpall}
```

After xparse may have been loaded …

```
12754 \AtBeginDocument{
```

… and *only* if xcolor was loaded:

```
12755 \IfPackageLoadedTF{xcolor}{
12756 \LWR@traceinfo{patching xcolor}
```

The print version:

`\colorboxBlock` `\colorboxBlock` is the same as `\colorbox`:

```
12757 \LetLtxMacro\colorboxBlock\colorbox
```

The original definition is reused by the new versions:

```
12758 \LetLtxMacro\LWR@orig@print@fcolorbox\fcolorbox
```

`\fcolorbox` `[`⟨*framemodel*⟩`]` `{`⟨*framecolor*⟩`}` `[`⟨*boxmodel*⟩`]` `{`⟨*boxcolor*⟩`}` `{`⟨*text*⟩`}`

In print mode, `\fcolorbox` is modified to accept a background color of none.

(`\fcolorbox` is particular about its optional arguments, thus the elaborate combinations of `\ifthenelse`.)

```
12759 \newsavebox{\LWR@colorminipagebox}
12760
12761 \NewDocumentCommand{\LWR@print@fcolorbox}{o m o m +m}{%
12762     \LWR@traceinfo{LWR@print@fcolorbox #2 #4}%
```

Pre-load the contents into an LR box so that they can be used inside a `\fcolorbox`:

```
12763     \begin{lrbox}{\LWR@colorminipagebox}%
12764     #5%
12765     \end{lrbox}%
```

Sort out the various optional arguments and the background color of none. In each case, the LRbox is placed inside a `\fcolorbox`.

The current color is remembered, then set to the frame, then the current color is used for the contents.

```
12766     \ifstrequal{#4}{none}%
12767     {% #4 none
12768         \LWR@traceinfo{background is none}%
12769         {% scope the \colorlet
12770             \colorlet{LWR@currentcolor}{.}%
12771             \color{#2}%
12772             \fbox{%
12773                 \color{LWR@currentcolor}%
12774                 \usebox{\LWR@colorminipagebox}%
12775             }% fbox
12776         }% colorlet
12777     }% #4 none
12778     {% #4 not none
12779     \LWR@traceinfo{background not none}%
```

```
    \IfValueTF{#1}%
    {%
        \IfValueTF{#3}%
     {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
     {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
    }%
    {% no value #1
        \IfValueTF{#3}%
     {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
       {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
    }% no value #1
    }% #4 not none
    \LWR@traceinfo{LWR@print@fcolorbox done}%
}

\renewrobustcmd*{\fcolorbox}{\LWR@print@fcolorbox}%
```

`\fcolorboxBlock` `[`⟨*framemodel*⟩`]` `{`⟨*framecolor*⟩`}` `[`⟨*boxmodel*⟩`]` `{`⟨*boxcolor*⟩`}` `{`⟨*text*⟩`}`

In print mode, `\fcolorboxBlock` is the same as `\fcolorbox`.

```
\newcommand*{\LWR@print@fcolorboxBlock}{\LWR@print@fcolorbox}

\newrobustcmd*{\fcolorboxBlock}{\LWR@print@fcolorboxBlock}
```

`fcolorminipage` (*env.*) `[`⟨*1:framemodel*⟩`]` `{`⟨*2:framecolor*⟩`}` `[`⟨*3:boxmodel*⟩`]` `{`⟨*4:boxcolor*⟩`}` `[`⟨*5:align*⟩`]` `[`⟨*6:height*⟩`]` `[`⟨*7:inner-align*⟩`]` `{`⟨*8:width*⟩`}`

In print mode, becomes a `\fcolorbox` containing a `minipage`:

```
\NewDocumentEnvironment{fcolorminipage}{o m o m O{c} O{} o m}
{%
    \LWR@traceinfo{*** fcolorminipage: #2 #4 #8}%
```

Pre-load the contents into an LR box so that they can be used inside a `\fcolorbox`:

```
    \begin{lrbox}{\LWR@colorminipagebox}%
```

If inner alignment is not given, use the outer alignment instead:

```
    \IfValueTF{#7}%
    {\begin{minipage}[#5][#6][#7]{#8}}%
    {\begin{minipage}[#5][#6][#5]{#8}}%
}%
{%
    \end{minipage}%
    \end{lrbox}%
    \LWR@traceinfo{*** starting end fcolorminipage #1 #2 #3 #4 #8}%
```

Sort out the various optional arguments and the background color of `none`. In each case, the LRbox is placed inside a `\fcolorbox`.

The current color is remembered, then set to the frame, then the current color is used for the contents.

```
    \ifstrequal{#4}{none}%
    {% #4 none
```

```
        {% scope the \colorlet
            \colorlet{LWR@currentcolor}{.}%
            \color{#2}%
            \fbox{%
                \color{LWR@currentcolor}%
                \usebox{\LWR@colorminipagebox}%
            }% fbox
        }% colorlet
    }% #4 none
    {% #4 not none
        \IfValueTF{#1}%
        {%
        \IfValueTF{#3}%
     {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
     {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
        }%
        {% no value #1
        \IfValueTF{#3}%
     {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
       {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
        }% no value #1
    }% #4 not none
    \LWR@traceinfo{*** finished end fcolorminipage}%
}
```

xcolor is known to have been loaded, and provided HTML versions of the following, and the print versions are provide above, so now they may be `\LW@formatted`.

```
\LWR@formatted{colorbox}
\LWR@formatted{colorboxBlock}
\LWR@formatted{fcolorbox}
\LWR@formatted{fcolorboxBlock}
\LWR@formattedenv{fcolorminipage}

\LWR@traceinfo{xcolor patches done}
}{}% xcolor loaded
}% AtBeginDocument

\end{warpall}
```

90 chemmacros environments

`\makepolymerdelims` and redox reactions must be enclosed in a `lateximage` during HTML output. These environments are provided here in print mode, and in the chemmacros code in HTML mode, as a high-level semantic syntax which automatically embeds the contents in a `lateximage` with an appropriate `alt` tag.

for PRINT output:

```
\begin{warpprint}

\AtBeginDocument{
\IfPackageLoadedTF{chemmacros}{
```

Env `polymerdelims`

```
\DeclareDocumentEnvironment{polymerdelims}{}
    {}{}
```

Env redoxreaction {⟨*space above*⟩} {⟨*space below*⟩}

For print output, extra space is include above and below the image, and a `latexmage` is not necessary. This extra space must be enforced, even inside a float, so zero-width rules are used.

For the HTML version, see section 193.5.

```
\DeclareDocumentEnvironment{redoxreaction}{m m}
    {\rule{0pt}{#1}}{\rule[-#2]{0pt}{#2}}

}{}% chemmacros
}% AtBeginDocument

\end{warpprint}
```

91 cleveref

loading order

cleveref and lwarp-cleveref with its associated macro patches are automatically preloaded at the end of the preamble via `\AtEndPreamble` and `\AfterEndPreamble`. This is done because the HTML conversion requires cleveref. The user's document may not require cleveref, thus the user may never explicitly load it, so during HTML output lwarp loads it last. If the user's document preamble uses cleveref options, or functions such as `\crefname`, then cleveref may be loaded in the user's preamble near the end, and lwarp's additional loading of cleveref will have no effect.

`\AtEndPreable` forces cleveref to be loaded last, if it has not yet been loaded by the user.

for HTML output:

```
\begin{warpHTML}

\AtEndPreamble{
    \RequirePackage{cleveref}
}

\end{warpHTML}
```

92 Preexisting label and reference definitions

Remember and patch some label-related defintions. These will be further encased and patched by other packages later.

`\label` and `\pageref` do NOT change their behavior according to print or HTML output, and thus do not use the `\LWR@formatted` system.

for HTML output:

```
\begin{warpHTML}

\LetLtxMacro\LWR@orig@label\label% includes memoir, before cleveref
\LetLtxMacro\label\LWR@new@label

\LetLtxMacro\LWR@orig@pageref\pageref
\LetLtxMacro\pageref\LWR@new@pageref

\end{warpHTML}
```

93 picture environment

picture (*env.*) The picture environment is enclosed inside a \lateximage.

for HTML output:

```
\begin{warpHTML}
```

picture (*env.*)

```
\BeforeBeginEnvironment{picture}{\begin{lateximage}[picture]}

\AfterEndEnvironment{picture}{\end{lateximage}}
```

```
\end{warpHTML}
```

94 Minipages and Boxes

A css flexbox is used for minipages and parboxes, allowing external and internal vertical positioning.

⚠ inline — A line of text with an inline minipage or \parbox will have the minipage or \parbox placed onto its own line, because a paragraph is a block element and cannot be made inline-block.

placement — minipages and \parboxes will be placed side-by-side in HTML unless you place a \newline between them.

side-by-side — Side-by-side minipages may be separated by \quad, \qquad, \enskip, \hspace, \hfill, or a \rule. When inside a center environment, the result is similar in print and HTML. Paragraph tags are suppressed between side-by-side minipages and these spacing commands, but not at the start or end of the paragraph.

⚠ minipage in a span — There is limited support for minipages inside an HTML . An HTML cannot appear inside a . While in a , minipages, and \parboxes, and any enclosed lists have limited HTML tags, resulting in an "inline" format, without markup except for HTML breaks. Use \newline or \par for an HTML break.

⚠ minipage size — When using minipage, \parbox, and fminipage, a virtual 6×9 inch text area is used for \linewidth, \textwidth, and \textheight, both for sizing the minipage, and also for its contents.

if width is \linewidth — If a minipage or \parbox is assigned a width of exactly \linewidth, in HTML it is automatically given no HTML width, thus allowed to fill the line as needed, similar to how it appears in print output.

full-width if HTML — A new macro \minipagefullwidth requests that, during HTML output, the next single minipage or \parbox be generated without an HTML width attribute, allowing it to be the full width of the display rather than the declared print-output width. This may be useful where the printed version's width makes no sense in HTML.

⚠ tabular, multicols — Inside a tabular or multicols environment, where the width depends on the browser window, \minipagefullwidth is effectively used by default for every minipage or \parbox inside the environment.

\UseMinipageWidths
\IgnoreMinipageWidths — \UseMinipageWidths may be used to tell lwarp to honor the specified widths of all following minipages and

\parboxes until the end of the local scope, and \IgnoreMinipageWidths may be used to tell lwarp to ignore the specified widths.

⚠ multicol Inside a multicols, \linewidth is divided by the specified number of columns.

⚠ text alignment Nested minipages adopt their parent's text alignment in HTML, whereas in regular LaTeX PDF output they do not. Use a flushleft or similar environment in the child minipage to force a text alignment.

for HTML output:
```
\begin{warpHTML}
```

94.1 Computed lengths

\LWR@minipagewidth (*Len*) Used to convert the width into printable units.

```
\newlength{\LWR@minipagewidth}
```

\LWR@minipageheight (*Len*) Used to convert the height into printable units.

```
\newlength{\LWR@minipageheight}
```

94.2 Virtual page size

LWR@virtualpagedepth (*Ctr*) Used to only reset the line width at the outermost minipage.

```
\newcounter{LWR@virtualpagedepth}
\setcounter{LWR@virtualpagedepth}{0}
```

LWR@setvirtualpage (*env.*) * [⟨*columns*⟩]

If not nesting a minipage, adjust \linewidth, \textwidth, and \textheight for a virtual 6×9 page, and start on a new PDF page to help prevent page overflows.

If starred, force a new page in the PDF before generating more HTML. This may be done to reduce the chance of page overflow when starting a new minipage.

The optional number of columns defaults to 1.

```
\NewDocumentEnvironment{LWR@setvirtualpage}{s O{1}}{%
    \ifnumequal{\value{LWR@virtualpagedepth}}{0}{%
        \IfBooleanT{#1}{\LWR@maybe@orignewpage}%
        \setlength{\linewidth}{6in/#2}%
        \setlength{\textwidth}{6in}%
        \setlength{\textheight}{9in}%
    }{}%
    \addtocounter{LWR@virtualpagedepth}{1}%
}
{\addtocounter{LWR@virtualpagedepth}{-1}}
```

94.3 Footnote handling

Also see section 60 for other forms of footnotes. Minipage footnotes are gathered in section 60.5, and then placed into the document in section 94.4.

94.4 Minipage handling

LWR@minipagefullwidth (*bool*) Should the next minipage have no HTML width?

```
\newbool{LWR@minipagefullwidth}
\boolfalse{LWR@minipagefullwidth}
```

LWR@forceminipagefullwidth (*bool*) Should the next minipage have no HTML width? Used to force full width for all minipages in an environment such as tabular or multicols, where the actual width depends on the browser width. Controlled by \useminipagewidths and \ignoreminipagewidths.

```
\newbool{LWR@forceminipagefullwidth}
\boolfalse{LWR@forceminipagefullwidth}
```

\minipagefullwidth Requests that the next minipage have no width tag in HTML:

for HTML output:

```
\newcommand*{\minipagefullwidth}{\global\booltrue{LWR@minipagefullwidth}}
```

\UseMinipageWidths Locally requests that minipage widths be honored.

```
\newcommand*{\UseMinipageWidths}{\boolfalse{LWR@forceminipagefullwidth}}
```

\IgnoreMinipageWidths Locally requests that minipage widths be ignored.

```
\newcommand*{\IgnoreMinipageWidths}{\booltrue{LWR@forceminipagefullwidth}}
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\minipagefullwidth}{}
\newcommand*{\UseMinipageWidths}{}
\newcommand*{\IgnoreMinipageWidths}{}
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

LWR@minipagethispar (*bool*) Has a minipage been seen this paragraph? If true, prevents paragraph tags around horizontal space between minipages.

```
\newbool{LWR@minipagethispar}
\boolfalse{LWR@minipagethispar}
```

LWR@minipage@depth (*Ctr*) Used to track whether to change footnote styles in a lateximage inside an HTML minipage.

```
\newcounter{LWR@minipage@depth}
\setcounter{LWR@minipage@depth}{0}
```

LWR@mpfootnote@store (*Ctr*) Used to maintain minipage footnote number while nesting inside a lateximage.

```
\newcounter{LWR@mpfootnote@store}
```

minipage (*env.*) [⟨*vert position*⟩] [⟨*height*⟩] [⟨*inner vert position*⟩] {⟨*width*⟩}

The vertical positions may be 'c', 't', or 'b'. The inner position may also be 's'.

When using \linewidth, \textwidth, or \textheight, these are scaled proportionally to a 6×9 inch text area.

```
\NewDocumentEnvironment{LWR@HTML@sub@minipage}{m m m m}
{%
\LWR@traceinfo{minipage}%
```

Start an environment, in which width and height is computed based on a virtual page size instead of the extra-large PDF page used during HTML tag generation.

```
\begin{LWR@setvirtualpage}*%
```

Save the requested width now that \linewidth, etc. are adjusted to virtual size.

```
\setlength{\LWR@minipagewidth}{#4}%
\ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
    \addtolength{\LWR@minipagewidth}{3em}% room for frames
}{}%
\LWR@traceinfo{computed width is \LWR@printlength{\LWR@minipagewidth}}%
```

Compute height:

```
\setlength{\LWR@minipageheight}{\textheight}% default unless specified
\ifblank{#2}{}{\setlength{\LWR@minipageheight}{#2}}%
```

LaTeX wants to start a paragraph for the virtual minipage, then start a paragraph again for the contents of the minipage, so cancel the paragraph tag handling until the minipage has begun.

```
\ifbool{FormatWP}{\newline}{}%
\LWR@stoppars%
```

If FormatWP, add a text frame:

```
\ifbool{FormatWP}{%

\addtocounter{LWR@thisautoidWP}{1}%
\LWR@htmltag{%
    div id=\textquotedbl%
        \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
    \textquotedbl\ % space
    class=\textquotedbl{}wpminipage\textquotedbl%
}%

}{}%
```

Create the tag with optional alignment style:

```
\LWR@traceinfo{minipage: creating div class}%
\LWR@htmltag{div class=\textquotedbl{}minipage\textquotedbl\ style=\textquotedbl%
\ifthenelse{\equal{#1}{t}}{\LWR@print@mbox{vertical-align:bottom} ; }{}%
\ifthenelse{\equal{#1}{c}}{\LWR@print@mbox{vertical-align:middle} ; }{}%
\ifthenelse{\equal{#1}{b}}{\LWR@print@mbox{vertical-align:top} ; }{}%
\ifthenelse{\equal{#3}{t}}{\LWR@print@mbox{justify-content:flex-start} ; }{}%
\ifthenelse{\equal{#3}{c}}{\LWR@print@mbox{justify-content:center} ; }{}%
```

```
\ifthenelse{\equal{#3}{b}}{\LWR@print@mbox{justify-content:flex-end} ; }{}%
\ifthenelse{\equal{#3}{s}}{\LWR@print@mbox{justify-content:space-between} ; }{}%
```

Print the width and optional height styles:

```
\LWR@traceinfo{minipage: about to print the width of \LWR@printlength{\LWR@minipagewidth}}%
\ifbool{LWR@minipagefullwidth}%
{\global\boolfalse{LWR@minipagefullwidth}}%
{%
    \ifbool{LWR@forceminipagefullwidth}%
        {}%
        {%
            \ifdimequal{#4}{\linewidth}%
                {}%
                {width:\LWR@printlength{\LWR@minipagewidth} ; }%
        }%
}%
\LWR@traceinfo{minipage: about to print the height}%
\ifblank{#2}{}{height:\LWR@printlength{\LWR@minipageheight} ; }%
\textquotedbl%
}%
```

Finish with an empty line to start the contents on a new line.

```

% The preceding empty line is required.
```

Set the user-accessible line and text width and height values inside the virtual minipage. These do not affect the actual size of the PDF output, but are used by any reference to `\linewidth`, etc. inside the virtual minipage being created here. `\LWR@minipagewidth` was the original then padded by 3em, which is restored here. This is done instead of settings back to #4, in case #4 was `\linewidth`, which was changed to 6in above.

```
\ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
    \addtolength{\LWR@minipagewidth}{-3em}% undo frame padding
}{}%
\setlength{\linewidth}{\LWR@minipagewidth}%
```

`\raggedright` cancels hyphenation, which will be done by HTML instead.

```
\LWR@print@raggedright%


    \LWR@newautopagelabel{page}%
```

Set minipage footnotes:

```
\def\@mpfn{mpfootnote}%
\def\thempfn{\thempfootnote}\c@mpfootnote\z@%
\let\@footnotetext\@mpfootnotetext%
```

Track depth for `lateximage` footnote type:

```
\addtocounter{LWR@minipage@depth}{1}%
```

Resume paragraph tag handling for the contents of the minipage:

```
\LWR@startpars%
\ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%

=== begin minipage ===

}{}%
\LWR@traceinfo{minipage: finished starting the minipage}%
}% finished \minipage
{% \endminipage
```

Print pending minipage footnotes:

```
\LWR@printpendingmpfootnotes%
```

End the environment with closing tag:

```
\ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%

=== end minipage ===

}{}%
\LWR@stoppars%

\ifbool{FormatWP}{%

\LWR@htmlelementend{div}%

}{}%
```

Wrapup:

```
\addtocounter{LWR@minipage@depth}{-1}%

\LWR@htmldivclassend{minipage}%

\end{LWR@setvirtualpage}%
\LWR@startpars%
\ifbool{FormatWP}{\newline}{}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
\global\booltrue{LWR@minipagethispar}%
\LWR@traceinfo{LWR@minipage: done}%
}

\NewDocumentEnvironment{LWR@HTML@minipage}{O{t} O{} O{t} m}
    {\LWR@HTML@sub@minipage{#1}{#2}{#3}{#4}}
    {\endLWR@HTML@sub@minipage}

\LWR@formattedenv{minipage}
```

94.5 \parbox, \mbox, \makebox, \framebox, \fbox, \raisebox

for HTML output:

\parbox [⟨*pos*⟩] [⟨*height*⟩] [⟨*inner-pos*⟩] {⟨*width*⟩} {⟨*text*⟩}

A parbox uses the minipage code:

```
\NewDocumentCommand{\LWR@HTML@parbox}{O{t} O{} O{t} m +m}
{
\LWR@traceinfo{parbox of width #4}%
\begin{minipage}[#1][#2][#3]{#4}%
#5
\end{minipage}%
}

\LWR@formatted{parbox}
```

`\mbox` {⟨*text*⟩} Nullified for HTML.

```
\newcommand*{\LWR@HTML@mbox}[1]{{#1}}

\LWR@formatted{mbox}
```

`\LWR@@makebox@paren` {⟨*width*⟩} , {⟨*height*⟩}

Adds to the style in `\LWR@temptwo`.

```
\NewDocumentCommand{\LWR@@makebox@paren}{m m}{%
\IfValueTF{#2}{%
    \setlength{\LWR@tempwidth}{#1\unitlength}%
    \setlength{\LWR@tempheight}{#2\unitlength}%
    \appto{\LWR@temptwo}{%
        \LWR@print@mbox{width:\LWR@printlength{\LWR@tempwidth}} ; % space
        \LWR@print@mbox{height:\LWR@printlength{\LWR@tempheight}} ; % space
    }%
}{%
    \PackageError{lwarp}%
        {(width,height) is missing a comma ',' character}%
        {\protect\makebox\space and \protect\framebox\space accept
            a size in the format (width,height).}%
}%
}
```

`\LWR@@makebox@align` {⟨*alignment character*⟩}

Adds to the style in `\LWR@temptwo`.

```
\newcommand*{\LWR@@makebox@align}[1]{%
    \def\LWR@align{center}%
    \ifstrequal{#1}{l}{\def\LWR@align{left}}{}%
    \ifstrequal{#1}{r}{\def\LWR@align{right}}{}%
    \ifstrequal{#1}{s}{\def\LWR@align{justify}}{}%
    \appto{\LWR@temptwo}{%
        \LWR@print@mbox{text-align:\LWR@align} ; %
    }%
}
```

`\makebox` (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@HTML@makebox}{>{\SplitArgument{1}{,}}d() o o +m}{%
```

Build the style depending on arguments:

```
    \begin{LWR@setvirtualpage}%
        \def\LWR@temptwo{}%
        \IfValueTF{#1}%
        {% (width,height) ..
            \LWR@@makebox@paren #1%
            \IfValueT{#2}%
            {% (width,height) [posn]
                \LWR@@makebox@align{#2}%
            }%
        }%
        {% [width]
            \IfValueT{#2}% [width]
            {%
                \setlength{\LWR@tempwidth}{#2}%
                \ifdimgreater{\LWR@tempwidth}{0pt}{%
                    \appto{\LWR@temptwo}{%
                        width:\LWR@printlength{\LWR@tempwidth} ; % space
                    }%
                }{}%
            }%
        }%
        \IfValueT{#3}%
        {% [width] [posn]
            \LWR@@makebox@align{#3}%
        }%
        \InlineClass[%
            \LWR@print@mbox{display:inline-block} ; %
            \LWR@temptwo%
        ]%
        {makebox}%
        {#4}%
    \end{LWR@setvirtualpage}%
}
\LWR@formatted{makebox}
```

`\framebox` (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@HTML@framebox}{d() o o +m}{%
    \fbox{\makebox(#1)[#2][#3]{#4}}%
}

\LWR@formatted{framebox}
```

`\LWR@forceminwidth` {⟨*legth*⟩}

Sets `\LWR@atleastonept` to be at least 1pt.

```
\newlength{\LWR@atleastonept}

\newcommand*{\LWR@forceminwidth}[1]{%
\setlength{\LWR@atleastonept}{#1}%
\ifthenelse{%
    \lengthtest{\LWR@atleastonept>0pt}\AND%
    \lengthtest{\LWR@atleastonept<1pt}%
}%
    {\setlength{\LWR@atleastonept}{1pt}}%
    {}%
```

```
}
```

`\LWR@fboxstyle` Prints the HTML attributes for a black border and padding.

`\LWR@forceminwidth` must be used first in order to set the border width.

```
\newcommand*{\LWR@fboxstyle}{%
\LWR@findcurrenttextcolor%
\LWR@traceinfo{LWR@fboxstyle B}%
border:\LWR@printlength{\LWR@atleastonept} solid \LWR@origpound\LWR@tempcolor ; %
padding:\LWR@printlength{\fboxsep} ; %
color:\LWR@origpound\LWR@tempcolor%
}
```

`\fbox` {⟨*text*⟩}

Creates a framed inline span enclosing the text.

Create a new HTML version, but don't use it until after xcolor may have loaded:

```
\newcommand{\LWR@HTML@fbox}[1]{%
    \LWR@traceinfo{HTML fbox}%
    \LWR@forceminwidth{\fboxrule}%
    \LWR@traceinfo{HTML fbox B}%
    \InlineClass[%
        \LWR@print@mbox{display:inline-block} ; %
        \LWR@fboxstyle%
    ]{fbox}{#1}%
    \LWR@traceinfo{HTML fbox: done}%
}
```

xcolor `\let`s things to `\fbox` when it is loaded, and this must remain even for HTML output while in a `lateximage`, so `\fbox` is not modified until `\AtBeginDocument`:

```
\AtBeginDocument{\LWR@formatted{fbox}}
```

`\fboxBlock` {⟨*text*⟩} Creates a framed HTML `<div>` of the text.

First, a print-mode version. This is newly defined for print mode, so it is defined inside `warpall`.

for HTML & PRINT:

```
\end{warpHTML}

\begin{warpall}
\let\fboxBlock\fbox
\end{warpall}

\begin{warpHTML}
```

for HTML output: Next, an HTML version:

```
\newcommand{\LWR@HTML@fboxBlock}[1]{%
\LWR@forceminwidth{\fboxrule}%
\LWR@stoppars%
\begin{BlockClass}[\LWR@fboxstyle]{fboxBlock}
#1
\end{BlockClass}
\LWR@startpars%
```

```
}

\LWR@formatted{fboxBlock}

\end{warpHTML}
```

`fminipage` (*env.*) `[`⟨*align*⟩`]` `[`⟨*height*⟩`]` `[`⟨*align*⟩`]` `{`⟨*width*⟩`}`

Creates a framed HTML around its contents.

for HTML & PRINT: Print version:

```
\begin{warpall}

\newsavebox{\LWR@fminipagebox}

\NewDocumentEnvironment{fminipage}{O{t} o O{t} m}
{%
```

An outer minipage will be used for vertical alignment. An inner minipage will be framed with \fbox.

If the optional inner alignment is not given, use the outer instead:

```
\IfValueTF{#3}%
{\def\LWR@thisalign{#3}}
{\def\LWR@thisalign{#1}}%
```

Form the outer minipage depending on whether a height was given. Make the outer minipage larger to compensate for the frame.

```
\IfValueTF{#2}%
{\minipage[#1][#2+2\fboxsep+2\fboxrule][\LWR@thisalign]{#4+2\fboxsep+2\fboxrule}}%
{\minipage[#1]{#4+2\fboxsep+2\fboxrule}}%
```

Capture the contents of the environment:

```
\begin{lrbox}{\LWR@fminipagebox}%
```

Nest the contents inside an inner minipage of the desired size:

```
\IfValueTF{#2}%
{\minipage[#1][#2][\LWR@thisalign]{#4}}%
{\minipage[#1]{#4}}%
}
{%
```

Close the inner minipage and the LR box with the contents:

```
\endminipage%
\end{lrbox}%
```

Create a frame around the contents of the environment:

```
\fbox{\usebox{\LWR@fminipagebox}}%
```

The entire thing is placed inside the outer minipage:

```
\endminipage%
}
\end{warpall}
```

HTML version:

for HTML output:

```
\begin{warpHTML}

\NewDocumentEnvironment{LWR@HTML@fminipage}{O{t} o O{t} m}
{%
\LWR@traceinfo{fminipage #1 #2 #3 #4}%
```

Locally change to the virtual page size before processing the requested sizes:

```
\begin{LWR@setvirtualpage}*%
\setlength{\LWR@tempwidth}{#4}%
\IfValueT{#2}{\setlength{\LWR@tempheight}{#2}}%
```

Use a rule of at least one pixel in width:

```
\LWR@forceminwidth{\fboxrule}%


\LWR@stoppars%


\begin{BlockClass}[%
\LWR@fboxstyle ; %
\IfValueT{#2}{height:\LWR@printlength{\LWR@tempheight} ; }%
\ifbool{LWR@minipagefullwidth}%
{\global\boolfalse{LWR@minipagefullwidth}}%
{%
    \ifbool{LWR@forceminipagefullwidth}%
        {}%
        {%
            \ifdimequal{\LWR@tempwidth}{\linewidth}%
                {}%
                {width:\LWR@printlength{\LWR@tempwidth} ; }%
        }%
}%
]{fminipage}%
}
{%
\end{BlockClass}%
\end{LWR@setvirtualpage}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
\global\booltrue{LWR@minipagethispar}%
\LWR@traceinfo{fminipage done}%
}
\LWR@formattedenv{fminipage}
```

`\raisebox` `{`⟨*raiselen*⟩`}` `[`⟨*height*⟩`]` `[`⟨*depth*⟩`]` `{`⟨*text*⟩`}`

```
\NewDocumentCommand{\LWR@HTML@raisebox}{m o o m}{%
#4%
```

```
}

\LWR@formatted{raisebox}

\end{warpHTML}
```

95 Direct formatting

⚠ \bfseries, etc. `\textbf`, etc. are supported, but `\bfseries`, etc. work only in some situations.

⚠ HTML special chars &, <, and > have special meanings in HTML. If `\&`, `\textless`, and `\textgreater` are used, proper HTML entities will be used, but there may be HTML parsing problems if these special characters occur unescaped in program listings or other verbatim text.

program listings For program listings, the listings package is supported, and its `literate` option is used to convert &, <, and > to proper HTML entities.

verbatim The various `verbatim`-related environments do not convert &, <, and >, so care must be taken to avoid accidentally including valid HTML code inside these environments. Adding a space on either side may be sufficient.

For high-level block and inline custom CSS classes, see section 52.10.

for HTML & PRINT:

```
\begin{warpall}
```

`FixSmallCaps` (*bool*) User may set `FixSmallCaps` to `true` if small caps are being incorrectly rendered as all caps.

```
\newbool{FixSmallCaps}
\boolfalse{FixSmallCaps}

\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

`\emph {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@emph}[1]{%
    {%
        \LWR@HTML@itshape%
        \LWR@htmlspan{em}{#1}%
    }%
}

\LWR@formatted{emph}
```

`\textmd {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@textmd}[1]{%
    {%
        \LWR@HTML@mdseries%
        \InlineClass(font-weight:normal){textmd}{#1}%
    }%
```

```
}

\LWR@formatted{textmd}
```

`\textbf {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@textbf}[1]{%
    {%
        \LWR@HTML@bfseries%
        \LWR@htmlspan{b}{#1}%
    }%
}

\LWR@formatted{textbf}
```

`\texteb {⟨text⟩}` From nfssext-cfr.

```
\IfPackageLoadedTF{nfssext-cfr}{
\DeclareRobustCommand{\LWR@HTML@texteb}[1]{%
    {%
        \LWR@HTML@ebweight%
        \InlineClass{texteb}{#1}%
    }%
}

\LWR@formatted{texteb}
}{% if not loaded
    \providerobustcmd{\texteb}[1]{}
}
```

`\textlg {⟨text⟩}` From nfssext-cfr.

```
\IfPackageLoadedTF{nfssext-cfr}{
\DeclareRobustCommand{\LWR@HTML@textlg}[1]{%
    {%
        \LWR@HTML@lgweight%
        \InlineClass{textlg}{#1}%
    }%
}

\LWR@formatted{textlg}
}{% if not loaded
    \providerobustcmd{\textlg}[1]{}
}
```

`\textrm {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@textrm}[1]{%
    {%
        \LWR@HTML@rmfamily%
        \InlineClass(font-family:serif){textrm}{#1}%
    }%
}

\LWR@formatted{textrm}
```

\textsf {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsf}[1]{%
    {%
        \LWR@HTML@sffamily%
        \InlineClass(font-family:sans){textsf}{#1}%
    }%
}

\LWR@formatted{textsf}
```

\texttt {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@texttt}[1]{%
    {%
        \LWR@HTML@ttfamily%
        \LWR@htmlspan{kbd}{#1}%
    }%
}

\LWR@formatted{texttt}
```

\textup {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textup}[1]{%
    {%
        \LWR@HTML@upshape%
        \InlineClass(font-style:normal){textup}{#1}%
    }%
}

\LWR@formatted{textup}
```

\textit {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textit}[1]{%
    {%
        \LWR@HTML@itshape%
        \LWR@htmlspan{i}{#1}%
    }%
}

\LWR@formatted{textit}
```

\textsc {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsc}[1]{%
    {%
        \LWR@HTML@scshape%
        \InlineClass{textsc}{#1}%
    }%
}

\LWR@formatted{textsc}
```

\textulc {⟨*text*⟩} From fontaxes.

```
\DeclareRobustCommand{\LWR@HTML@textulc}[1]{%
    {%
        \LWR@HTML@ulcshape%
        \InlineClass{textulc}{#1}%
    }%
}

\LWR@formatted{textulc}
```

\textsi {⟨*text*⟩}

```
\@ifundefined{textsi}{
    \LetLtxMacro\LWR@print@textsi\LWR@print@textsc
}{}

\DeclareRobustCommand{\LWR@HTML@textsi}[1]{%
    {%
        \LWR@HTML@sishape%
        \textsc{\textit{#1}}%
%         \InlineClass(
%             font-style: italic;
%             font-variant: small-caps ;
%             font-variant-numeric: oldstyle-nums ;
%         ){textsi}{#1}%
    }%
}

\LWR@formatted{textsi}
```

\textsl {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsl}[1]{%
    {%
        \slshape%
        \InlineClass(font-style:oblique){textsl}{#1}%
    }%
}

\LWR@formatted{textsl}
```

\textssc {⟨*text*⟩}

```
\newrobustcmd{\LWR@HTML@textssc}[1]{\textsc{#1}}
\LWR@formatted{textssc}
```

\textnormal {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textnormal}[1]{%
        \LWR@HTML@mdseries%
        \LWR@HTML@rmfamily%
        \LWR@HTML@upshape%
        \LWR@HTML@ulcshape%
        \InlineClass(%
            font-weight: normal;
            font-family: serif;
            font-style: normal;
            font-variant: normal;
```

```
            font-variant-numeric: normal ;
        ){textnormal}{#1}%
}

\LWR@formatted{textnormal}

\FilenameNullify{%
    \LetLtxMacro\emph\@firstofone%
    \LetLtxMacro\textmd\@firstofone%
    \LetLtxMacro\textbf\@firstofone%
    \LetLtxMacro\texteb\@firstofone%
    \LetLtxMacro\textlg\@firstofone%
    \LetLtxMacro\textrm\@firstofone%
    \LetLtxMacro\textsf\@firstofone%
    \LetLtxMacro\texttt\@firstofone%
    \LetLtxMacro\textup\@firstofone%
    \LetLtxMacro\textit\@firstofone%
    \LetLtxMacro\textsc\@firstofone%
    \LetLtxMacro\textulc\@firstofone%
    \LetLtxMacro\textsi\@firstofone%
    \LetLtxMacro\textsl\@firstofone%
    \LetLtxMacro\textssc\@firstofone%
    \LetLtxMacro\textnormal\@firstofone%
}
```

Remembers the current font family, series, and shape. fontaxes support is integrated here.

```
\newcommand*{\LWR@f@family}{rm}
\newcommand*{\LWR@f@series}{md}
\newcommand*{\LWR@f@shape}{up}
\newcommand*{\LWR@f@shapecaps}{ulc}
```

`\LWR@textcurrentfont {`⟨*text*⟩`}`

Prints the text with the current font choices. Avoids nesting repeated font selections.

```
\newcounter{LWR@textcurrentfontdepth}
\setcounter{LWR@textcurrentfontdepth}{0}

\newcommand*{\LWR@textcurrentfont}[1]{%
    \ifnumcomp{\value{LWR@textcurrentfontdepth}}{>}{0}%
        {%
            \addtocounter{LWR@textcurrentfontdepth}{1}%
            #1%
            \addtocounter{LWR@textcurrentfontdepth}{-1}%
        }%
        {%
            \addtocounter{LWR@textcurrentfontdepth}{1}%
            \ifboolexpr{%
                test {\ifdefstring{\LWR@f@family}{rm}} and
                test {\ifdefstring{\LWR@f@series}{md}} and
                test {\ifdefstring{\LWR@f@shape}{up}} and
                test {\ifdefstring{\LWR@f@shapecaps}{ulc}}
            }%
                {\InlineClass{textnormal}{#1}}%
                {%
                    \InlineClass{%
```

```
                                text\LWR@f@family\LWR@origtilde{}%
                                text\LWR@f@series\LWR@origtilde{}%
                                text\LWR@f@shape\LWR@origtilde{}%
                                text\LWR@f@shapecaps%
                            }%
                            {#1}%
                    }%
                \addtocounter{LWR@textcurrentfontdepth}{-1}%
            }%
}
```

`LWR@blocktextcurrentfont` (*env.*) Prints the contents with the current font choices.

```
\newenvironment*{LWR@blocktextcurrentfont}{%
\LWR@stoppars%
\BlockClass{%
        text\LWR@f@family\LWR@origtilde{}%
        text\LWR@f@series\LWR@origtilde{}%
        text\LWR@f@shape\LWR@origtilde{}%
        text\LWR@f@shapecaps%
    }%
}{\endBlockClass\LWR@startpars}
```

`\mdseries`

```
\newrobustcmd*{\LWR@HTML@mdseries}{%
    \LWR@print@mdseries%
    \renewcommand*{\LWR@f@series}{md}%
}
\LWR@formatted{mdseries}
```

`\bfseries`

```
\newrobustcmd*{\LWR@HTML@bfseries}{%
    \LWR@print@bfseries%
    \renewcommand*{\LWR@f@series}{bf}%
}
\LWR@formatted{bfseries}
```

`\ebweight` From nfssext-cfr.

```
\IfPackageLoadedTF{nfssext-cfr}{
\newrobustcmd*{\LWR@HTML@ebweight}{%
    \LWR@print@ebweight%
    \renewcommand*{\LWR@f@series}{eb}%
}
\LWR@formatted{ebweight}
}{}
```

`\lgweight` From nfssext-cfr.

```
\IfPackageLoadedTF{nfssext-cfr}{
\newrobustcmd*{\LWR@HTML@lgweight}{%
    \LWR@print@lgweight%
    \renewcommand*{\LWR@f@series}{lg}%
```

```
}
\LWR@formatted{lgweight}
}{}
```

\rmfamily

```
\newrobustcmd*{\LWR@HTML@rmfamily}{%
    \LWR@print@rmfamily%
    \renewcommand*{\LWR@f@family}{rm}%
}
\LWR@formatted{rmfamily}
```

\sffamily

```
\newrobustcmd*{\LWR@HTML@sffamily}{%
    \LWR@print@sffamily%
    \renewcommand*{\LWR@f@family}{sf}%
}
\LWR@formatted{sffamily}
```

\ttfamily

```
\newrobustcmd*{\LWR@HTML@ttfamily}{%
    \LWR@print@ttfamily%
    \renewcommand*{\LWR@f@family}{tt}%
}
\LWR@formatted{ttfamily}
```

The following use \AtBeginDocument due to the LaTeX core \reinstall@nfss@defs, which redefines these \AtBeginDocument. See **texdoc source2e**.

\upshape

```
\newrobustcmd*{\LWR@HTML@upshape}{%
    \LWR@print@upshape%
    \renewcommand*{\LWR@f@shape}{up}%
}
\AtBeginDocument{\LWR@formatted{upshape}}
```

\itshape

```
\newrobustcmd*{\LWR@HTML@itshape}{%
    \LWR@print@itshape%
    \renewcommand*{\LWR@f@shape}{it}%
}
\AtBeginDocument{\LWR@formatted{itshape}}
```

\scshape Note: \LWR@print@scshape is not used here since some fonts, such as erewhon, copy/paste as all-caps.

```
\newrobustcmd*{\LWR@HTML@scshape}{%
    \ifbool{FixSmallCaps}{}{%
        \LWR@print@scshape%
    }%
```

```
    \renewcommand*{\LWR@f@shapecaps}{sc}%
}
\AtBeginDocument{\LWR@formatted{scshape}}
```

`\ulcshape` From fontaxes.

```
\@ifundefined{ulcshape}{
    \LetLtxMacro\ulcshape\upshape
}{}
\newrobustcmd*{\LWR@HTML@ulcshape}{%
    \LWR@print@ulcshape%
    \renewcommand*{\LWR@f@shapecaps}{ulc}%
}
\AtBeginDocument{\LWR@formatted{ulcshape}}
```

`\sishape`

```
\@ifundefined{sishape}{
    \LetLtxMacro\sishape\scshape
}{}
\newrobustcmd*{\LWR@HTML@sishape}{%
    \ifbool{FixSmallCaps}{}{%
        \LWR@print@sishape%
    }%
    \renewcommand*{\LWR@f@shape}{it}
    \renewcommand*{\LWR@f@shapecaps}{sc}%
}
\AtBeginDocument{\LWR@formatted{sishape}}
```

`\slshape`

```
\newrobustcmd*{\LWR@HTML@slshape}{%
    \LWR@print@slshape%
    \renewcommand*{\LWR@f@shape}{sl}%
}
\AtBeginDocument{\LWR@formatted{slshape}}
```

`\sscshape`

```
\newrobustcmd{\LWR@HTML@sscshape}{\LWR@HTML@scshape}
\AtBeginDocument{\LWR@formatted{sscshape}}
```

`\normalfont`

```
\newrobustcmd*{\LWR@HTML@normalfont}{\rmfamily\mdseries\upshape\ulcshape}
\LWR@formatted{normalfont}

\FilenameNullify{%
    \LetLtxMacro\rmfamily\@empty%
    \LetLtxMacro\sffamily\@empty%
    \LetLtxMacro\ttfamily\@empty%
    \LetLtxMacro\bfseries\@empty%
    \LetLtxMacro\ebweight\@empty%
    \LetLtxMacro\lgweight\@empty%
    \LetLtxMacro\mdseries\@empty%
    \LetLtxMacro\upshape\@empty%
```

```
    \LetLtxMacro\slshape\@empty%
    \LetLtxMacro\sishape\@empty%
    \LetLtxMacro\scshape\@empty%
    \LetLtxMacro\itshape\@empty%
    \LetLtxMacro\ulcshape\@empty%
    \LetLtxMacro\sscshape\@empty%
    \LetLtxMacro\normalfont\@empty%
}
```

`\sp` {⟨*text*⟩}

For siunitx. Must work in math mode.

```
\renewcommand{\sp}[1]{\text{<sup>#1</sup>}{}}
```

`\sb` {⟨*text*⟩}

For siunitx. Must work in math mode.

```
\renewcommand{\sb}[1]{\text{<sub>#1</sub>}{}}
```

`\textsuperscript` {⟨*text*⟩}

```
\newrobustcmd{\LWR@HTML@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
\LWR@formatted{textsuperscript}
```

`\@textsuperscript` {⟨*text*⟩}

```
\newcommand{\LWR@HTML@@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
\LWR@formatted{@textsuperscript}
```

`\textsubscript` {⟨*text*⟩}

```
    \newrobustcmd{\LWR@HTML@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
    \LWR@formatted{textsubscript}
```

`\@textsubscript` {⟨*text*⟩}

```
    \newcommand{\LWR@HTML@@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
    \LWR@formatted{@textsubscript}
```

`\up` {⟨*text*⟩} Prints superscript.

This is `\let` at the beginning of the document in case some other package has changed the definition.

```
\AtBeginDocument{\let\up\textsuperscript}
```

`\fup` {⟨*text*⟩} Prints superscript.

Supports fmtcount package.

This is `\let` at the beginning of the document in case some other package has changed the definition.

```
\AtBeginDocument{\let\fup\textsuperscript}
```

`\underline {⟨text⟩}`

```
\renewcommand{\underline}[1]{%
    \InlineClass%
        (text-decoration:underline; text-decoration-skip: auto)%
        {underline}{#1}%
}
```

`\LWR@overline {⟨text⟩}`

```
\newcommand{\LWR@overline}[1]{%
    \InlineClass%
        (text-decoration:overline; text-decoration-skip: auto)%
        {overline}{#1}%
}
```

`\LWR@currenttextcolor` The color to use for text and `\rule`, defaulting to black:

```
\newcommand*{\LWR@currenttextcolor}{black}
```

`\LWR@tempcolor` `\LWR@tempcolortwo` `\LWR@tempcolorthree` The color converted to HTML colorspace.

```
\newcommand*{\LWR@tempcolor}{}
\newcommand*{\LWR@tempcolortwo}{}
\newcommand*{\LWR@tempcolorthree}{}
```

`\LWR@findcurrenttextcolor` Sets `\LWR@tempcolor` to the current color.

```
\newcommand*{\LWR@findcurrenttextcolor}{%
    \renewcommand{\LWR@tempcolor}{000000}%
}
```

`\LWR@textcurrentcolor {⟨text⟩}` Like `\textcolor` but uses the current `\color` instead.

```
\NewDocumentCommand{\LWR@textcurrentcolor}{m}{%
    \renewcommand*{\LWR@currenttextcolor}{black}%
    #1%
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

`\LWR@textcurrentfont {⟨text⟩}`

Prints the text with the current font choices.

```
\newcommand*{\LWR@textcurrentfont}[1]{#1}
```

`LWR@blocktextcurrentfont` (*env.*) Prints the contents with the current font choices.

```
\newenvironment*{LWR@blocktextcurrentfont}{}{}
```

`\FilenameNullify {`⟨*macros to nullify*⟩`}`

```
\newcommand*{\FilenameNullify}[1]{}

\end{warpprint}
```

96 Skips, spaces, font sizes

for HTML output:

```
\begin{warpHTML}
```

`\,` and `\thinspace` may be redefined by other packages, so are redefined `\AtBeginDocument` here.

Direct-formatting space commands become HTML entities:

```
\AtBeginDocument{%
\renewrobustcmd*{\,}{\HTMLunicode{202f}}% HTML thin non-breakable space, not using LWR@formatted
\newrobustcmd*{\LWR@HTML@thinspace}{\HTMLunicode{202f}}% HTML thin non-breakable space
\LWR@formatted{thinspace}
\newrobustcmd*{\LWR@HTML@negthinspace}{\HTMLunicode{202f}}   % HTML thin non-breakable space
\LWR@formatted{negthinspace}
\renewrobustcmd*{~}{\HTMLentity{nbsp}}% cannot use \LWR@formatted
\newrobustcmd*{\LWR@HTML@textellipsis}{\HTMLunicode{2026}}
\LWR@formatted{textellipsis}
\newrobustcmd*{\LWR@HTML@vdots}{\HTMLunicode{22EE}}
\LWR@formatted{vdots}
}
```

Direct-formatting font sizes are remembered for future use:

```
\newcommand*{\LWR@font@size}{normalsize}

\newrobustcmd*{\LWR@HTML@normalsize}{\renewcommand*{\LWR@font@size}{normalsize}}
\LWR@formatted{normalsize}

\newrobustcmd*{\LWR@HTML@small}{\renewcommand*{\LWR@font@size}{small}}
\LWR@formatted{small}

\newrobustcmd*{\LWR@HTML@footnotesize}{\renewcommand*{\LWR@font@size}{footnotesize}}
\LWR@formatted{footnotesize}

\newrobustcmd*{\LWR@HTML@scriptsize}{\renewcommand*{\LWR@font@size}{scriptsize}}
\LWR@formatted{scriptsize}

\newrobustcmd*{\LWR@HTML@tiny}{\renewcommand*{\LWR@font@size}{tiny}}
\LWR@formatted{tiny}

\newrobustcmd*{\LWR@HTML@large}{\renewcommand*{\LWR@font@size}{large}}
\LWR@formatted{large}

\newrobustcmd*{\LWR@HTML@Large}{\renewcommand*{\LWR@font@size}{Large}}
\LWR@formatted{Large}

\newrobustcmd*{\LWR@HTML@LARGE}{\renewcommand*{\LWR@font@size}{LARGE}}
\LWR@formatted{LARGE}

```

```
\newrobustcmd*{\LWR@HTML@huge}{\renewcommand*{\LWR@font@size}{huge}}
\LWR@formatted{huge}

\newrobustcmd*{\LWR@HTML@Huge}{\renewcommand*{\LWR@font@size}{Huge}}
\LWR@formatted{Huge}


\DeclareDocumentCommand{\onecolumn}{}{}

\DeclareDocumentCommand{\twocolumn}{O{}}{
#1

}
```

\hfill

```
\newcommand*{\LWR@HTML@hfill}{\qquad}
\LWR@formatted{hfill}
```

\hrulefill

```
\newcommand*{\LWR@HTML@hrulefill}{%
    \ifbool{LWR@doingapar}%
        {\rule{1in}{1pt}}%
        {%
            \LWR@findcurrenttextcolor%
            \ifdefstring{\LWR@tempcolor}{000000}%
            {%
                \begin{BlockClass}{hrule}%
                \end{BlockClass}%
            }%
            {%
                \begin{BlockClass}[%
                  border-top: 1px solid \LWR@origpound\LWR@tempcolor % space
                ]{hrule}%
                \end{BlockClass}%
            }%
        }%
}%
\LWR@formatted{hrulefill}
```

\dotfill

```
\newcommand*{\LWR@HTML@dotfill}{\dots}
\LWR@formatted{dotfill}
```

\newpage

```
\renewcommand*{\newpage}{

}
```

\newline Uses the HTML <br /> element.

```
\newrobustcmd*{\LWR@newlinebr}{\unskip\LWR@htmltag{br /}\LWR@orignewline}%
\LetLtxMacro\newline\LWR@newlinebr
```

\\ Redefined to \LWR@endofline or \LWR@tabularendofline.

\LWR@endofline * [⟨*len*⟩]

\\ is assigned to \LWR@endofline at \LWR@LwarpStart.

Inside tabular, \\ is temporarily changed to \LWR@tabularendofline.

```
\LetLtxMacro\LWR@origendofline\\
\NewDocumentCommand{\LWR@endofline}{s O{0pt}}
{%
\newline%

\setlength{\LWR@templengthone}{#2}%
\ifdimgreater{\LWR@templengthone}{0pt}{\newline}{}%
}
```

\LWR@minipagestartpars Minipages are often placed side-by-side inside figures, with a bit of horizontal space to separate them. Since HTML does not allow a to be inside a p, paragraphs must be turned off during the generation of the minipage, then turned on after the minipage is complete. When this occurs between side-by-side minipages, lwarp correctly suppresses the paragraph tags between the minipages, unless some other text is between the minipages. Such text forms its own paragraph, resulting in text after a minipage to be on its own line. Since people often place small horizontal space between minipages, it is desirable to maintain this space if possible. lwarp tries to do this by remembering that a minipage has been seen, in which case paragraph tags are suppressed around \hspace, \enskip, \quad, and \qquad until the end of the paragraph, when the closing p tag is created.

\hspace
\enskip
\quad
\qquad

When a minipage is seen, the boolean LWR@minipagethispar is set, telling the following horizontal whitespace commands to try to suppress their surrounding paragraph tags. LWR@minipagethispar is cleared at the next end of paragraph, when the HTML paragraph closing tag is generated.

Placed just before \hspace, \quad, or \qquad's HTML output.

```
\newcommand*{\LWR@minipagestartpars}{%
    \ifbool{LWR@minipagethispar}{\LWR@startpars}{}%
}
```

\LWR@minipagestoppars Placed just after \hspace, \quad, or \qquad's HTML output.

```
\newcommand*{\LWR@minipagestoppars}{%
    \ifbool{LWR@minipagethispar}{\LWR@stoppars}{}%
}
```

\quad Handles special minipage & horizontal space interactions. Uses 2003 EM SPACE to pass validation.

```
\newrobustcmd*{\LWR@HTML@quad}{%
    \LWR@minipagestoppars%
    \HTMLunicode{2003}%
    \LWR@minipagestartpars%
}
\LWR@formatted{quad}
```

\qquad Handles special minipage & horizontal space interactions.

```
\newrobustcmd*{\LWR@HTML@qquad}{\quad\quad}
\LWR@formatted{qquad}
```

\enskip Handles special minipage & horizontal space interactions.

```
\newrobustcmd*{\LWR@HTML@enskip}{%
    \LWR@minipagestoppars%
    \HTMLunicode{2002}%
    \LWR@minipagestartpars%
}
\LWR@formatted{enskip}
```

\LWR@tempwidth (*Len*)
\LWR@tempheight (*Len*)
\LWR@tempraise (*Len*)

Used to compute span width, height, raise for \hspace and \rule:

```
\newlength{\LWR@tempwidth}
\newlength{\LWR@tempheight}
\newlength{\LWR@tempraise}
```

\hspace * {⟨*length*⟩} * {⟨*length*⟩}

Handles special minipage & horizontal space interactions.

Prints a span of a given width. Ignores the optional star.

\hspace{\fill} is converted to \hspace{2em}, equal to \qquad.

```
\NewDocumentCommand{\LWR@HTML@hspace}{s m}{%
\setlength{\LWR@tempwidth}{#2}%
```

If \fill, change to \qquad:

```
\ifnum\gluestretchorder\LWR@tempwidth>0%
\setlength{\LWR@tempwidth}{2em}%
\fi%
```

Only if the width is greater than zero:

```
\ifdimcomp{\LWR@tempwidth}{>}{0pt}{%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
    \LWR@minipagestoppars%
```

Support the HTML thin wrappable space:

```
    \ifdimcomp{\LWR@tempwidth}{=}{.16667em}%
    {%
        \HTMLunicode{2009}% thin breakable space
    }%
```

Print the span with the converted width. Not rounded.

```
    {%
```

```
        \LWR@htmltagc{%
        span style=\textquotedbl{}width:\LWR@printlength{\LWR@tempwidth}; % extra space
                display:inline-block\textquotedbl%
        }%
```

If formatting for a word processor, approximate with a number of \quads, in case a span of a given width is not supported:

```
        \ifbool{FormatWP}{%
            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
                \quad%
                \addtolength{\LWR@templengthone}{-1em}%
            }%
        }%
```

If NOT formatting for a word processor, include an empty comment to avoid an empty span:

```
        {\LWR@htmlcomment{}}%
```

Close the span:

```
        \LWR@htmltagc{/span}%
    }%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
    \LWR@minipagestartpars%
}{}% width greater than 0
}%
\LWR@formatted{hspace}
```

`\LWR@vspace` * {⟨*length*⟩} Nullified vspace.

```
\NewDocumentCommand{\LWR@HTML@vspace}{s m}{}

\LWR@formatted{vspace}
```

`\linebreak` [⟨*num*⟩] Inserts an HTML `br` tag.

```
\renewcommand*{\linebreak}[1][]{\newline}
```

`\nolinebreak` [⟨*num*⟩]

```
\renewcommand*{\nolinebreak}[1][]{}
```

`\pagebreak` [⟨*num*⟩] Starts a new paragraph.

```
\renewcommand*{\pagebreak}[1][]{

}
```

`\nopagebreak` [⟨*num*⟩]

```
13695 \renewcommand*{\nopagebreak}[1][]{}
```

`\enlargethispage` * {⟨*len*⟩}

```
13696 \RenewDocumentCommand{\enlargethispage}{s m}{}
```

`\clearpage`
`\cleardoublepage`

```
13697 \renewcommand*{\clearpage}{}
13698 \renewcommand*{\cleardoublepage}{}
```

`\rule` [⟨*raise*⟩] {⟨*width*⟩} {⟨*height*⟩}

Handles special minipage & horizontal space interactions.

Creates a span of a given width and height. Ignores the optional star.

`\fill` is zero-width, so `\hspace{\fill}` is ignored.

```
13699 \newcommand*{\LWR@HTML@rule}[3][]{%
```

The width is copied into a temporary LaTeX length, from which comparisons and conversions may be made:

```
13700 \setlength{\LWR@tempwidth}{#2}%
```

If it's zero-width then skip the entire rule:

```
13701 \ifthenelse{\lengthtest{\LWR@tempwidth=0pt}}%
13702 {}% zero- width
13703 {% non-zero width
```

If it's non-zero width, set a minimal thickness so that it more reliably shows in the browser:

```
13704     \ifthenelse{%
13705         \lengthtest{\LWR@tempwidth>0pt}\AND%
13706         \lengthtest{\LWR@tempwidth<1pt}%
13707     }%
13708         {\setlength{\LWR@tempwidth}{1pt}}%
13709         {}%
```

Likewise with height:

```
13710     \setlength{\LWR@tempheight}{#3}%
13711     \ifthenelse{%
13712         \lengthtest{\LWR@tempheight>0pt}\AND%
13713         \lengthtest{\LWR@tempheight<1pt}%
13714     }%
13715         {\setlength{\LWR@tempheight}{1pt}}%
13716         {}%
```

If had a minipage this paragraph, try to inline the rule without generating paragraph tags:

```
13717     \LWR@minipagestoppars%
```

Print the span with the converted width and height. The width and height are NOT rounded, since a height of less than 1pt is quite common in LaTeX code.

```
    \LWR@findcurrenttextcolor%
    \LWR@htmltagc{%
    span\LWR@indentHTML%
    style=\textquotedbl%
```

The HTML background color is used to draw the filled rule according to the LaTeX foreground color set by `\textcolor`.

```
    \ifbool{FormatWP}{}{background:\LWR@currenttextcolor ; }%
```

The width and height are printed, converted to PT:

```
    width:\LWR@printlength{\LWR@tempwidth} ; %
    height:\LWR@printlength{\LWR@tempheight} ; %
```

The raise height is converted to a CSS transform. The `*2` raise multiplier is to approximately match HTML output's X height. Conversion to a LaTeX length allows a typical LaTeX expression to be used as an argument for the raise, whereas printing the raise argument directly to HTML output without conversion to a LaTeX length limits the allowable syntax. To do: A superior method would compute a ratio of LaTeX ex height, then print that to HTML with an ex unit.

```
    \ifblank{#1}%
    {}%
    {%
        \setlength{\LWR@tempraise}{0pt-#1}%
        \setlength{\LWR@tempraise}{\LWR@tempraise*2}%
        \LWR@indentHTML%
        -ms-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
        \LWR@indentHTML%
       -webkit-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
        \LWR@indentHTML%
        transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
        \LWR@indentHTML%
    }%
```

Display inline-block to place the span inline with the text:

```
    display:inline-block;\textquotedbl\LWR@orignewline%
    }%
```

If formatting for a word processor, approximate with a number of underscores, in case a span of a given width is not supported:

```
        \ifbool{FormatWP}{%
            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
                \_{}%
                \addtolength{\LWR@templengthone}{-1em}%
            }%
        }%
```

If NOT formatting for a word processor, add a comment to avoid an empty :

```
13747         {\LWR@htmlcomment{}}%
```

Close the span:

```
13748     \LWR@htmltagc{/span}%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
13749     \LWR@minipagestartpars%
13750 }% non-zero width
13751 }
13752
13753 \LWR@formatted{rule}

13754 \end{warpHTML}
```

97 \phantomsection

for HTML output:

```
13755 \begin{warpHTML}
```

\LWR@phantomsection Emulate the hyperref \phantomsection command, often used to insert the bibliography into the table of contents. Ignores \ForceHTMLTOC.

```
13756 \newrobustcmd*{\LWR@phantomsection}{%
13757     \begingroup%
13758     \boolfalse{LWR@forcinghtmltoc}%
13759     \section*{}%
13760     \endgroup%
13761 }

13762 \end{warpHTML}
```

98 \LaTeX and other logos

Logos for HTML and print modes:

Some of these logos may be redefined in a later package, so after loading other packages, and at the beginning of the document, their definitions are finally set by \LWR@formatted.

For CSS conversions, see:
http://edward.oconnor.cx/2007/08/tex-poshlet
http://nitens.org/taraborelli/texlogo
and the spacing described in the metafont package documentation.

for HTML & PRINT:

```
13763 \begin{warpall}
13764 \newbool{LWR@warnXe}
13765 \boolfalse{LWR@warnXe}
13766
13767 \newrobustcmd*{\Xe}
13768     {%
```

```
        X\hspace{-.1667em}\raisebox{-.5ex}{E}%
        \global\booltrue{LWR@warnXe}%
    }

\AtBeginDocument{
    \IfPackageLoadedTF{graphics}{
        \IfPackageLoadedTF{metalogo}{}{
            \renewrobustcmd*{\Xe}
                {X\hspace{-.1667em}\raisebox{-.5ex}{\reflectbox{E}}}
        }
    }{}
}

\AtEndDocument{
    \ifbool{LWR@warnXe}{
        \PackageNoteNoLine{lwarp}{Load graphicx or graphics
            for improved XeTeX logo}
    }{}
}

\providerobustcmd*{\XeTeX}{\mbox{\Xe\hspace{-.125em}\TeX}}
\providerobustcmd*{\XeLaTeX}{\mbox{\Xe\hspace{-.125em}\LaTeX}}
\providerobustcmd*{\AmS}{%
    \leavevmode\hbox{$\mathcal A\kern-.2em\lower.376ex%
    \hbox{$\mathcal M$}\kern-.2em\mathcal S$}%
}
\newrobustcmd*{\LyX}{\textsf{LyX}}
\providerobustcmd*{\LuaTeX}{\mbox{Lua\TeX}}
\providerobustcmd*{\LuaLaTeX}{\mbox{Lua\LaTeX}}
\providerobustcmd*{\BibTeX}{\mbox{B\textsc{ib}\TeX}}
\providerobustcmd*{\MakeIndex}{\mbox{\textit{MakeIndex}}}
\providerobustcmd*{\ConTeXt}{\mbox{Con\TeX{}t}}
\providerobustcmd*{\MiKTeX}{\mbox{MiK\TeX}}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

The print-mode versions of the following may be changed by metalogo, so their print formatting is recorded `\AtBeginDocument`.

`\TeX` TEX

`latexlogo` is a css class used to properly typeset the E and A in LATEX and friends.

`latexlogofont` is a css class used to select the font for the rest of the logo in LATEX, LuaTEX, ConTEXt, etc.

```
\newrobustcmd*{\LWR@HTML@TeX}
{%
    \InlineClass{latexlogofont}%
    {%
        \InlineClass{latexlogo}%
        {%
            T%
            \InlineClass{latexlogosub}{e}%
            X%
        }%
    }%
}
```

```
\AtBeginDocument{\LWR@formatted{TeX}}% may have been patched by metalogo
```

\LaTeX LaTeX, LaTeX2ε
\LaTeXe

```
\newrobustcmd*{\LWR@HTML@LaTeX}
{%
    \InlineClass{latexlogofont}%
    {%
        \InlineClass{latexlogo}%
        {%
            L%
            \InlineClass{latexlogosup}{a}%
            T%
            \InlineClass{latexlogosub}{e}%
            X%
        }%
    }%
}

\AtBeginDocument{\LWR@formatted{LaTeX}}% may have been patched by metalogo


\newrobustcmd*{\LWR@HTML@LaTeXe}
{%
    \LaTeX%
    \InlineClass{latexlogofont}{%
        \InlineClass{latexlogotwoe}{%
            2%
            \InlineClass{latexlogotwoesub}{\HTMLunicode{03B5}}%
        }%
    }%
}
\AtBeginDocument{\LWR@formatted{LaTeXe}}% may have been patched by metalogo
```

\LuaTeX LuaTeX, LuaLaTeX
\LuaLaTeX

```
\newrobustcmd*{\LWR@HTML@LuaTeX}{\InlineClass{latexlogofont}{Lua}\TeX}
\AtBeginDocument{\LWR@formatted{LuaTeX}}% may have been patched by metalogo

\newrobustcmd*{\LWR@HTML@LuaLaTeX}{\InlineClass{latexlogofont}{Lua}\LaTeX}
\AtBeginDocument{\LWR@formatted{LuaLaTeX}}% may have been patched by metalogo
```

\XeTeX XeTeX, XeLaTeX
\XeLaTeX

xetexlogo is a css class which aligns the backwards E in XeTeX and spaces TeX appropriately.

xelatexlogo is a css class which aligns the backwards E in XeLaTeX and spaces LaTeX appropriately.

```
\newrobustcmd*{\LWR@HTML@Xe}
    {%
        X%
        \InlineClass{xelatexlogosub}{\HTMLunicode{18e}}%
    }
\AtBeginDocument{\LWR@formatted{Xe}}% may have been patched by metalogo

\newrobustcmd*{\LWR@HTML@XeTeX}{\InlineClass{xelatexlogo}{\Xe}\TeX}
```

```
\AtBeginDocument{\LWR@formatted{XeTeX}}% may have been patched by metalogo

\newrobustcmd*{\LWR@HTML@XeLaTeX}{\InlineClass{xelatexlogo}{\Xe}\LaTeX}
\AtBeginDocument{\LWR@formatted{XeLaTeX}}% may have been patched by metalogo
```

\ConTeXt ConTEXt

```
\newrobustcmd*{\LWR@HTML@ConTeXt}{%
    \InlineClass{latexlogofont}{Con}\TeX{}%
    \InlineClass{latexlogofont}{t}%
}
\LWR@formatted{ConTeXt}
```

\BibTeX BIBTEX, *MakeIndex*
\MakeIndex

```
\newrobustcmd*{\LWR@HTML@BibTeX}
    {\InlineClass{latexlogofont}{B\textsc{ib}}\TeX}
\LWR@formatted{BibTeX}

\newrobustcmd*{\LWR@HTML@MakeIndex}
    {\InlineClass{latexlogofont}{\textit{MakeIndex}}}
\LWR@formatted{MakeIndex}
```

\AmS $\mathcal{AMS}$

amslogo is a css class used for the $\mathcal{AMS}$ logo.

```
\AtBeginDocument{%
\newrobustcmd*{\LWR@HTML@AmS}
{%
    \InlineClass{amslogo}{%
        \textit{%
            A%
            \InlineClass{latexlogosub}{M}%
            S%
        }%
    }%
}%
\LWR@formatted{AmS}
}
```

\MiKTeX MiKTEX

```
\newrobustcmd*{\LWR@HTML@MiKTeX}{\InlineClass{latexlogofont}{MiK}\TeX}
\LWR@formatted{MiKTeX}
```

\LyX LyX

lyxlogo is a css class used for the LyX logo.

```
\newrobustcmd*{\LWR@HTML@LyX}{\InlineClass{lyxlogo}{LyX}}
\LWR@formatted{LyX}

\end{warpHTML}
```

99 Starting and stopping lwarp

for HTML output:
```
\begin{warpHTML}
```

\LWR@LwarpStart
\LWR@LwarpEnd

Automatically sets up the HTML-related actions for the start and end of the document.

```
\AfterEndPreamble{\LWR@LwarpStart}
\AtEndDocument{\LWR@LwarpEnd}
\DeclareHookRule{enddocument}{lwarp}{after}{legacy}
```

```
\end{warpHTML}
```

100 Loading array

array is required for lwarp's column parsing. It and its patches are now loaded.

for HTML output:
```
\begin{warpHTML}
\RequirePackage{array}
```

The following are compared with the tabular preamble > to add css classes to adjust tabular cells. Defined here now that \arraybackslash is defined after array is loaded.

```
\edef\LWR@detect@centeringarraybackslash{\centering\arraybackslash}
\edef\LWR@detect@raggedrightarraybackslash{\raggedright\arraybackslash}
\edef\LWR@detect@raggedleftarraybackslash{\raggedleft\arraybackslash}
\def\LWR@detect@itshape{\itshape}
\def\LWR@detect@bfseries{\bfseries}
\def\LWR@detect@bfit{\bfseries\itshape}
\end{warpHTML}
```

101 Loading everyshi patches

everyshi is emulated by the LaTeX core, so its patches are loaded here. \AtBeginDocument is used in case an older verison of LaTeX is used.

for HTML output:
```
\begin{warpHTML}
\AtBeginDocument{
    \IfPackageLoadedTF{everyshi}{
        \RequirePackage{lwarp-everyshi}
    }{}
}
\end{warpHTML}
```

102 Loading textcomp patches

textcomp has now been integrated into the LaTeX core, so its patches are loaded now.

for HTML output:
```
\begin{warpHTML}
\RequirePackage{lwarp-textcomp}
\end{warpHTML}
```

103 Loading amsmath, amsthm patches, centernot

amsmath, amsthm, and centernot may have been preloaded, such as by newtx, so their patches are loaded now.

for HTML output:
```
\begin{warpHTML}
\IfPackageLoadedTF{amsthm}{
    \RequirePackage{lwarp-amsthm}
}{}

\IfPackageLoadedTF{amsmath}{
    \RequirePackage{lwarp-amsmath}
}{}
```

amsthm may load centernot, so centernot must be checked second.

```
\IfPackageLoadedTF{centernot}{
    \RequirePackage{lwarp-centernot}
}{}
\end{warpHTML}
```

104 Loading KOMA-SCRIPT class patches

Load patches to koma-script.

for HTML output:
```
\begin{warpHTML}

\IfClassLoadedTF{scrbook}{\RequirePackage{lwarp-patch-komascript}}{}
\IfClassLoadedTF{scrartcl}{\RequirePackage{lwarp-patch-komascript}}{}
\IfClassLoadedTF{scrreprt}{\RequirePackage{lwarp-patch-komascript}}{}

\end{warpHTML}
```

105 Loading MEMOIR class patches

Load patches to memoir.

for PRINT output:
```
\begin{warpprint}
\IfClassLoadedTF{memoir}{\LWR@origRequirePackage{xcolor}}{}
\end{warpprint}
```

for HTML output:
```
\begin{warpHTML}
\IfClassLoadedTF{memoir}{\RequirePackage{lwarp-patch-memoir}}{}
\end{warpHTML}
```

106 ut* class patches

Load patches to uj* and ut* classes, as well as ltj* classes.

for HTML output:
```
\begin{warpHTML}
\newcommand*{\LWR@patchujtclasses}{
```

uj/t does not use \partname

```
    \def\@partnameformat{}

    \def\@partcntformat##1{%
        \prepartname%
        \csname the##1\endcsname%
        \postpartname%
        \quad%
    }
    \@ifundefined{chapter}{}{
        \def\@chapcntformat##1{%
            \prechaptername%
            \csname the##1\endcsname%
            \postchaptername%
            \quad%
        }
    }
    \renewcommand*{\LWR@printchaptername}{}
```

Use decimal points instead of centered dots:

```
    \renewcommand{\thepart}{\@Roman\c@part}
    \@ifundefined{chapter}{
        \renewcommand{\thesection}{\@arabic\c@section}
    }{
        \renewcommand{\thechapter}{\@arabic\c@chapter}
        \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
    }
    \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
    \renewcommand{\thesubsubsection}{%
    \thesubsection.\@arabic\c@subsubsection}
    \renewcommand{\theparagraph}{%
    \thesubsubsection.\@arabic\c@paragraph}
    \renewcommand{\thesubparagraph}{%
    \theparagraph.\@arabic\c@subparagraph}
    \@ifundefined{chapter}{
        \renewcommand{\thefigure}{\@arabic\c@figure}
        \renewcommand{\thetable}{\@arabic\c@table}
    }{
        \renewcommand{\thefigure}{%
        \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@figure}
        \renewcommand{\thetable}{%
        \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@table}
    }
}

\IfClassLoadedTF{ujarticle}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ujbook}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ujreport}{\LWR@patchujtclasses}{}
```

```
\IfClassLoadedTF{utarticle}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{utbook}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{utreport}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjarticle}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjbook}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjreport}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjsarticle}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjsbook}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjsreport}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjskiyou}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjspf}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjtarticle}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjtbook}{\LWR@patchujtclasses}{}
\IfClassLoadedTF{ltjtreport}{\LWR@patchujtclasses}{}

\end{warpHTML}
```

107 CTEX patches

Patches for ctex and related classes, which are loaded before lwarp.

All CTEX classes and the ctex package seem to load ctexpatch, so its presence is used to decide whether to have lwarp patch CTEX.

for HTML output:

```
\begin{warpHTML}
```

`\AtBeginDocument` in case the user set `FileSectionNames` in the preamble.

```
\AtBeginDocument{
    \IfPackageLoadedTF{ctexpatch}{%
        \def\@partcntformat#1{%
            \LWR@isolate{\CTEX@partname}~%
            \CTEX@part@aftername%
        }%

        \def\@partnameformat{}

        \def\@chapcntformat#1{%
            \LWR@isolate{\CTEX@chaptername}~%
            \CTEX@chapter@aftername%
        }%

        \renewcommand*{\LWR@printchaptername}{}
    }{}
}

\end{warpHTML}
```

108 kotexutf patches

Patch for kotexutf, which is loaded before lwarp.

kotexutf's `\@setref` was conflicting with lwarp's cross references.

for HTML output:

```
\begin{warpHTML}
```

If kotexutf's version of `\@setref` is detected, it is reverted to the original.

```
\AtBeginDocument{
\IfPackageLoadedTF{kotexutf}{%
    \def\LWR@kotexutf@setref#1#2#3{%
        \@setref@dhucs@orig{#1}{#2}{#3}%
      \ifx#1\relax\else
        \bgroup
        \dhucs@make@cjkchar@null
        \edef\@temp{\expandafter#2#1}\global\josatoks\expandafter{\@temp}%
        \egroup
      \fi%
    }%

    \ifdefequal{\@setref}{\LWR@kotexutf@setref}{
        \let\@setref\@setref@dhucs@orig
    }{}
}{}
}

\end{warpHTML}
```

109 babel and polyglossia warnings

lwarp prints a message instructing the user how to avoid the following error.

(These are not `\PackageWarnings` because there may not be a problem.)

lwarp uses cleveref, which has some limitations when using polyglossia, possibly resulting in the error

```
! Undefined control sequence. . . .  \__hook begindocument
```

To test compatibility, add

```
\usepackage{cleveref}
```

near the end of the preamble (as the last package to be loaded), and try to compile the print version. It may be necessary to set

```
\setdefaultlanguage{english}
```

or some other language supported by cleveref, then select other languages using `\setotherlanguages`.

Once the print version works with cleveref and polyglossia, the HTML version should work as well using lwarp.

for HTML output:

```
\begin{warpHTML}
\AtBeginDocument{

\IfPackageLoadedTF{polyglossia}{
    \PackageNoteNoLine{lwarp}
    {%
        Polyglossia has been loaded.  Lwarp also uses cleveref.\MessageBreak
        See the cleveref documentation regarding\MessageBreak
        polyglossia support.  Some languages are not supported.\MessageBreak
```

```
        --- \MessageBreak
        If the error\MessageBreak
        \space\space Undefined control sequence ...
        \protect\__hook begindocument\MessageBreak
        occurs here, use the polyglossia macro:\MessageBreak
        \space\space\protect\setmainlanguage\protect{...\protect}
    }
}{
    \IfPackageLoadedTF{babel}{
        \PackageNoteNoLine{lwarp}
        {%
            Babel has been loaded.  Lwarp also uses cleveref.\MessageBreak
            See the cleveref documentation regarding\MessageBreak
            babel support.  Some languages are not supported%
        }
    }{}
}

}
\end{warpHTML}
```

110 MathJax warnings

`\LWR@mathjaxwarn` {⟨*packagename*⟩} {⟨*More text.*⟩}

Issue a warning that MathJax is emulated. To be done `\AtBeginDocument`.

```
\newcommand*{\LWR@mathjaxwarn}[2]{%
    \IfPackageLoadedTF{lwarp-#1}{%
        \ifblank{#2}{%
            \PackageWarningNoLine{lwarp}
                {%
                 Lwarp provides emulation for MathJax when used\MessageBreak
                    with the #1 package%
                }
        }{%
            \PackageWarningNoLine{lwarp}
                {%
                 Lwarp provides emulation for MathJax when used\MessageBreak
                    with the #1 package.\MessageBreak
                    #2%
                }
        }%
    }{}%
}

% \begin{macro}{\LWR@nomathjaxwarn} \marg{packagename} \marg{More text.}
%
% Issue a warning that \MathJax\ is not supported.
% To be done \cs{AtBeginDocument}.
%
% \changes{v0.894}{2020/12/22}{Warn if using packages not supported by \MathJax.}
% \changes{v0.895}{2021/01/08}{Improved \MathJax\ warning.}
%    \begin{macrocode}
\newcommand*{\LWR@nomathjaxwarn}[2]{%
    \IfPackageLoadedTF{lwarp-#1}{%
        \ifblank{#2}{%
```

```
            \PackageWarningNoLine{lwarp}
                {%
                 Lwarp does not provide MathJax support for #1.\MessageBreak
                    Use SVG math by removing the Lwarp mathjax option%
                }
        }{%
            \PackageWarningNoLine{lwarp}
                {%
                 Lwarp does not provide MathJax support for #1.\MessageBreak
                    #2%
                }
        }%
    }{}%
}
```

`\LWR@forceSVGmessage` {⟨*packagename*⟩}

```
\newcommand*{\LWR@forceSVGmessage}[1]{%
    SVG math output may be enabled for select math\MessageBreak
    expressions to preserve #1 visual\MessageBreak
    features for those particular expressions.\MessageBreak
    Before the chosen inline math, use \protect\inlinemathother\MessageBreak
    to begin using SVG math, and \protect\inlinemathnormal\MessageBreak
    afterward to resume using MathJax math.\MessageBreak
    Before display math, use \protect\displaymathother\MessageBreak
    to begin using SVG math, and use \protect\displaymathnormal\MessageBreak
    after to resume using MathJax for the following math.\MessageBreak
    Or, use SVG math for all expressions by removing\MessageBreak
    the mathjax option for the lwarp package%
}
```

If MathJax is being used, issue a warning for certain packages.

```
\AtBeginDocument{
    \ifbool{mathjax}{
        \LWR@nomathjaxwarn{aligned-overset}{}
        \LWR@nomathjaxwarn{amscdx}{\LWR@forceSVGmessage{amscdx}}
        \LWR@mathjaxwarn{arydshln}
            {In a math array, do not use the optional argument\MessageBreak
            for \protect\cdashline.\space\space
            Furthermore, \protect\cline\space is not\MessageBreak
            supported by MathJax}
        \LWR@nomathjaxwarn{autoaligne}{}
        \LWR@mathjaxwarn{autonum}
            {MathJax does not support equation+.\MessageBreak
            You may use the warpprint and warpHTML\MessageBreak
            environments to isolate the package load\MessageBreak
            and the equation+ environments}
        \LWR@mathjaxwarn{bigdelim}
            {Delimiters appear only of the first line}
        \LWR@nomathjaxwarn{boldtensors}{}
        \LWR@mathjaxwarn{booktabs}
            {\protect\cmidrule\space is not displayed}
        \LWR@mathjaxwarn{breqn}
            {Each environment becomes an SVG image}
        \LWR@mathjaxwarn{colortbl}
            {Colors are ignored in MathJax.\MessageBreak
            (Text mode tabular does support colortbl.)\MessageBreak
            \LWR@forceSVGmessage{colortbl}}
```

```
        \LWR@mathjaxwarn{delarray}{\LWR@forceSVGmessage{delarray}}
        \LWR@nomathjaxwarn{gauss}{\LWR@forceSVGmessage{gauss}}
        \LWR@mathjaxwarn{hhline}
            {A simple \protect\hline\space is used}
        \LWR@mathjaxwarn{isomath}
        {Some of the symbol font macros such as \protect\mathsfbfit\MessageBreak
             do not use a sans font because MathJax does not yet\MessageBreak
                have sans Greek.  Tensors may look like vectors%
            }
        \LWR@nomathjaxwarn{jkmath}{\LWR@forceSVGmessage{jkmath}}
        \LWR@mathjaxwarn{libertinust1math}
        {Some of the symbol font macros such as \protect\mathsfbfit\MessageBreak
             do not use a sans font because MathJax does not yet\MessageBreak
                have sans Greek.  Tensors may look like vectors%
            }
        \LWR@mathjaxwarn{mathtools}
            {See the Lwarp manual regarding the disallowspaces\MessageBreak
            and showonlyrefs options, the alignat environment,\MessageBreak
            and \protect\DeclarePairedDelimiter\space and related%
            }
        \LWR@mathjaxwarn{mathspec}
            {Double quotes are removed, even inside \protect\text}
        \LWR@mathjaxwarn{mismath}
          {MathJax does not support \cs{enumber}, \cs{inumber},\MessageBreak
        \protect\jnumber, \protect\pinumber, \protect\MathUp, \protect\MathIt,\MessageBreak
            \protect\MathNumbers, or \protect\MathNormal.\MessageBreak
        \protect\itpi\space is made available as a clone of \protect\pi.\MessageBreak
            Tensors are not sans serif%
            }
        \LWR@mathjaxwarn{multirow}
            {Multirow works as expected in text mode, but\MessageBreak
            limited emulation is provided for MathJax math.\MessageBreak
           \protect\multirow\space ignores all arguments except\MessageBreak
            the text}
        \LWR@mathjaxwarn{nicematrix}
            {Keys/values are ignored in MathJax.\MessageBreak
            \protect\Cdots, etc. do not span multiple cells.\MessageBreak
           AutoNiceMatrix, etc. are not supported for MathJax.\MessageBreak
        \protect\CodeBefore, \protect\Body, and \protect\CodeAfter\MessageBreak
            \space\space also are not supported for MathJax.\MessageBreak
            \LWR@forceSVGmessage{nicematrix}%
            }
        \LWR@nomathjaxwarn{pb-diagram}{\LWR@forceSVGmessage{pb-diagram}}
%         \LWR@mathjaxwarn{physics}
% %             {The third-party extension is not used.\MessageBreak
%             {The MathJax v3 extension is used.\MessageBreak
%              See the Lwarp manual for details}
        \LWR@mathjaxwarn{siunitx}
        {Place \protect\sisetup\space before \protect\begin{document}.\MessageBreak
            Many optional arguments are ignored}
        \LWR@nomathjaxwarn{tensind}{}
        \LWR@mathjaxwarn{unicode-math}
            {Do not use embedded Unicode characters.\MessageBreak
            (Not all characters are encoded correctly.)\MessageBreak
            Some symbol fonts are not supported by MathJax,\MessageBreak
            and are only approximated.\MessageBreak
        Greek macros such as \protect\alpha\space respond to the math-style\MessageBreak
            option.  Latin symbols does not, per MathJax\MessageBreak
          limitations, unless placed inside \protect\symbit\space or similar}
        \LWR@nomathjaxwarn{unitsdef}{}
```

```
14210         \LWR@mathjaxwarn{witharrows}
14211             {Arrows can only point to the next line.\MessageBreak
14212             Text is only placed on a single line}
14213         \LWR@nomathjaxwarn{xy}
14214          {In text, xy works as-is.  SVG images will be generated.\MessageBreak
14215             \LWR@forceSVGmessage{xy}}
14216     }{}
14217 }
```

File 2 **lwarp-2in1.sty**

§ 111 Package 2in1

2in1 (*Pkg*) 2in1 is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{2in1}
```

File 3 **lwarp-2up.sty**

§ 112 Package 2up

2up (*Pkg*) 2up is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{2up}[2010/05/15]

\def\source#1#2#3{}
\def\target#1#2#3{}
\def\targetlayout#1{}
\newdimen\pageseplength
\newdimen\pagesepwidth
\newdimen\pagesepoffset
\def\twoupemptypage{}
\def\twoupclearpage{}
\def\twoupeject{}
\def\twouparticle{}
\def\twoupplain{}
\def\twouplegaltarget{}
\def\twouplandscape{}
\def\TwoupWrites{}
```

File 4 **lwarp-a4.sty**

§ 113 Package a4

a4 (*Pkg*) a4 is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a4}[2004/04/15]

\newcommand*{\WideMargins}{}
```

File 5 **lwarp-a4wide.sty**

§ 114 Package a4wide

a4wide (*Pkg*) a4wide is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a4wide}[1994/08/30]
```

File 6 **lwarp-a5comb.sty**

§ 115 Package a5comb

a5comb (*Pkg*) a5comb is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a5comb}
```

File 7 **lwarp-abstract.sty**

§ 116 Package abstract

(*Emulates or patches code by* Peter Wilson.)

abstract (*Pkg*) abstract is supported and patched by lwarp.

⚠ missing TOC If using the number option with file splits, be sure to place the table of contents before the abstract. The number option causes a section break which may cause a file split, which would put a table of contents out of the home page if it is after the abstract.

for HTML output: memoir provides an abstract environment even though it is not an article or report class. Meanwhile, lwarp loads book to emulate memoir, but book does not have an abstract environment, so when the abstract package is loaded for emulation there is no pre-existing abstract to redefine, which would cause an error. Thus, a null abstract is provide here:

```
\ProvideDocumentEnvironment{abstract}{}{}{}
```

Accept all options for lwarp-abstract:

```
\LWR@ProvidesPackagePass{abstract}[2009/06/08]

\AtBeginDocument{
\BeforeBeginEnvironment{abstract}{
\LWR@forcenewpage
\BlockClass{abstract}
}
\AfterEndEnvironment{abstract}{\endBlockClass}
}

\renewcommand{\@bsrunintitle}{%
\hspace*{\abstitleskip}%
{\abstractnamefont%
\InlineClass{abstractrunintitle}{\abstractname}%
\@bslabeldelim}%
}

\IfClassLoadedTF{memoir}
{
  \renewenvironment{abstract}{%
```

```
% %     \titlepage
%     \null\vfil
%     \@beginparpenalty\@lowpenalty
  \setup@bstract
    \if@bsrunin
    \else
%       \if@bsstyle
%         \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
%       \else
        \ifnumber@bs
          \num@bs
        \else
          \begin{\absnamepos}%
  \abstractnamefont \BlockClassSingle{abstracttitle}{\abstractname}
%             \@endparpenalty\@M
          \end\absnamepos%
       \vspace{\abstitleskip}%
        \fi
%       \fi
%       \vspace{\abstitleskip}%
    \fi
    \put@bsintoc%
    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
    {\par\end{@bstr@ctlist}%\vfil\null%\endtitlepage
    }
}{% not memoir
\if@titlepage
  \renewenvironment{abstract}{%
%     \titlepage
    \null\vfil
    \@beginparpenalty\@lowpenalty
    \if@bsrunin
    \else
      \if@bsstyle
        \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
      \else
        \ifnumber@bs
          \num@bs
        \else
          \begin{\absnamepos}%
  \abstractnamefont \BlockClassSingle{abstracttitle}{\abstractname}
            \@endparpenalty\@M
          \end\absnamepos%
%%        \vspace{\abstitleskip}%
        \fi
      \fi
      \vspace{\abstitleskip}%
    \fi
    \put@bsintoc%
    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
    {\par\end{@bstr@ctlist}\vfil\null%\endtitlepage
    }
\else
  \renewenvironment{abstract}{%
    \if@bsrunin
    \else
      \if@bsstyle
        \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
      \else
        \ifnumber@bs
```

```
          \num@bs
        \else
\begin{\absnamepos}%
\abstractnamefont\BlockClassSingle{abstracttitle}{\abstractname}%
\end\absnamepos%
%%        \vspace{\abstitleskip}%
        \fi
      \fi
      \vspace{\abstitleskip}%
    \fi
    \put@bsintoc%
    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
    {\par\end{@bstr@ctlist}}
\fi
}% not memoir
```

File 8 **lwarp-academicons.sty**

§ 117 Package academicons

(*Emulates or patches code by* Diogo A. B. Fernandes.)

academicons (*Pkg*) academicons is patched for use by lwarp.

If \aiicon is used, the name of the icon is used in the alt tag. Otherwise, for each of the individual icon macros, a generic alt tag is used.

for HTML output:

```
\LWR@ProvidesPackagePass{academicons}[2018/06/27]

\LetLtxMacro\LWR@orig@symbol\symbol

\let\LWR@academicons@orig@AI\AI

\newcommand*{\LWR@academicons@symbol}[1]{%
    \begin{lateximage}*[academicon][academicons#1]%
    \begingroup%
    \LWR@academicons@orig@AI%
    \LWR@orig@symbol{#1}%
    \endgroup%
    \end{lateximage}%
}

\renewcommand*{\AI}{%
    \LetLtxMacro\symbol\LWR@academicons@symbol%
}

\renewcommand*{\aiicon}[1]
{%
    \begin{lateximage}*[#1 icon][academicons#1]%
    \AI\csname aiicon@#1\endcsname%
    \end{lateximage}%
}
```

File 9 **lwarp-accents.sty**

§ 118 Package

accents

(*Emulates or patches code by* Javier Bezos.)

accents (*Pkg*) accents is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{accents}[2006/05/12]
```

For MathJax:

```
\begin{warpMathJax}
\LWR@infoprocessingmathjax{accents}

\CustomizeMathJax{\newcommand{\ring}[1]{\mathring{#1}}}
\CustomizeMathJax{\newcommand{\accentset}[2]{\overset{#1{}}{#2}}}
```

As of this writing, MathJax v3 does not yet support groups for macros, so for \underaccent, the originals are remembered here, then they are temporarily redefined and used inside \underaccent, then restored to their originals. \LARGE gives a reasonable size, and \raise is used to adjust vertically without introducing extra line space.

```
\CustomizeMathJax{\let\LWRgrave\grave}
\CustomizeMathJax{\let\LWRacute\acute}
\CustomizeMathJax{\let\LWRcheck\check}
\CustomizeMathJax{\let\LWRbreve\breve}
\CustomizeMathJax{\let\LWRbar\bar}
\CustomizeMathJax{\let\LWRhat\hat}
\CustomizeMathJax{\let\LWRdot\dot}
\CustomizeMathJax{\let\LWRtilde\tilde}
\CustomizeMathJax{\let\LWRddot\ddot}
\CustomizeMathJax{\let\LWRvec\vec}
\CustomizeMathJax{\let\LWRwidetilde\widetilde}

\CustomizeMathJax{\newcommand{\underaccent}[2]{%
    {%
    \renewcommand{\grave}[1]{{\LARGE\LWRgrave{##1}}}%
    \renewcommand{\acute}[1]{{\LARGE\LWRacute{##1}}}%
    \renewcommand{\check}[1]{{\LARGE\LWRcheck{##1}}}%
    \renewcommand{\breve}[1]{{\LARGE\LWRbreve{##1}}}%
    \renewcommand{\bar}[1]{{\LARGE\LWRbar{##1}}}%
    \renewcommand{\hat}[1]{{\LARGE\LWRhat{##1}}}%
    \renewcommand{\dot}[1]{{\LARGE\LWRdot{##1}}}%
    \renewcommand{\tilde}[1]{{\LARGE\LWRtilde{##1}}}%
    \renewcommand{\ddot}[1]{{\LARGE\LWRddot{##1}}}%
    \renewcommand{\vec}[1]{{\LARGE\LWRvec{##1}}}%
    \renewcommand{\widetilde}[1]{{\LARGE\LWRwidetilde{\hphantom{#2}}}}%
    \underset{\raise 2pt {#1{}}}{#2}%
    \let\grave\LWRgrave%
    \let\acute\LWRacute%
    \let\check\LWRcheck%
    \let\breve\LWRbreve%
    \let\bar\LWRbar%
```

```
    \let\hat\LWRhat%
    \let\dot\LWRdot%
    \let\tilde\LWRtilde%
    \let\ddot\LWRddot%
    \let\vec\LWRvec%
    \let\widetilde\LWRwidetilde%
    }%
}}

\CustomizeMathJax{\newcommand{\undertilde}[1]{%
    \underset{\raise 3pt {\widetilde{\hphantom{#1}}}}{#1}%
}}
\end{warpMathJax}
```

File 10 **lwarp-accessibility.sty**

§ 119 Package accessibility

accessibility (*Pkg*) accessibility is emulated.

for HTML output: Discard all options for lwarp-accessibility:

```
\LWR@ProvidesPackageDrop{accessibility}[2019/10/14]

\newcommand{\alt}[1]{\ThisAltText{#1}}
\newcommand{\newhref}[3]{\ThisAltText{#2}\LWR@href{#1}{#3}}%
\providecommand{\thead}[1]{\textbf{#1}}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\alt}[1]{}}
\CustomizeMathJax{\newcommand{\thead}[1]{\text{\textbf{#1}}}}
\end{warpMathJax}
```

File 11 **lwarp-accsupp.sty**

§ 120 Package accsupp

accsupp (*Pkg*) accsupp is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{accsupp}[2018/03/28]

\newcommand*{\BeginAccSupp}[1]{}
\newcommand*{\EndAccSupp}[1]{}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\BeginAccSupp}[1]{}}
\CustomizeMathJax{\newcommand{\EndAccSupp}[1]{}}
\end{warpMathJax}
```

File 12 **lwarp-acro.sty**

§ 121 Package **acro**

(*Emulates or patches code by* Clemens Niederberger.)

acro (*Pkg*) acro is patched for use by lwarp.

⚠ formats Define acronymn formats using \textbf instead of \bfseries etc.

for HTML output:

```
\LWR@ProvidesPackagePass{acro}[2019/10/12]
```

\DeclareAcronym is used in the preamble, where lwarp has not yet made the dollar active, so temporarily enable lwarp math catcode just for this definition:

```
\ExplSyntaxOn
\NewDocumentCommand \LWR@DeclareAcronym {mm}
{
    \acro_declare_acronym:nn {#1} {#2}
    \catcode`\$=3% lwarp
}
\ExplSyntaxOff

\RenewDocumentCommand{\DeclareAcronym}{}{
    \catcode`\$=\active% lwarp
    \LWR@DeclareAcronym
}
```

Replace dot fill with simple dots:

```
\ExplSyntaxOn
\cs_new_protected:Npn \LWR@HTML@acro_dot_fill: {\dots\space}
\LWR@formatted{acro_dot_fill:}
\ExplSyntaxOff
```

Modified to activate the current font:

```
\ExplSyntaxOn
\IfPackageAtLeastTF{acro}{2020/04/29}%
{}% v3 or later
{% before v3
\IfPackageAtLeastTF{acro}{2019/09/23}%
{% v2.10 or later
\cs_gset_protected:Npn \__acro_typeset:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \use:x
        {
          \bool_if:cTF {l__acro_custom_#1_format_bool}
            { \exp_not:v {l__acro_custom_#1_format_tl} }
            { \exp_not:v {l__acro_#1_format_tl} }
            {\exp_not:N\LWR@textcurrentfont{#2}}%   lwarp
        }
    \group_end:
```

```
  }

\cs_gset_protected:Npn \__acro_ending_format:nn #1#2
  {
    \bool_if:NTF \l__acro_include_endings_format_bool
      {
        \str_case:nn {#1}
          {
            {long}
            {
              \bool_if:NTF \l__acro_custom_long_format_bool
                { \l__acro_custom_long_format_tl }
                {
                  \bool_if:NTF \l__acro_first_instance_bool
                    { \l__acro_first_long_format_tl }
                    { \l__acro_long_format_tl }
                }
            }
            {short}
            {
              \bool_if:NTF \l__acro_custom_short_format_bool
                { \l__acro_custom_short_format_tl }
                { \l__acro_short_format_tl }
            }
            {alt}
            {
              \bool_if:NTF \l__acro_custom_alt_format_bool
                { \l__acro_custom_alt_format_tl }
                { \l__acro_alt_format_tl }
            }
          }
      }
      { \use:n }
      {\exp_not:N\LWR@textcurrentfont{#2}}% lwarp
  }
}% v2.10 or later
{% before v2.10
\cs_gset_protected:Npn \acro_write_short:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \bool_if:NTF \l__acro_custom_format_bool
        { \l__acro_custom_format_tl }
        { \l__acro_short_format_tl }
      {\LWR@textcurrentfont{#2}}% lwarp
    \group_end:
  }

\cs_gset_protected:Npn \acro_write_alt:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \bool_if:NTF \l__acro_custom_format_bool
        { \l__acro_custom_format_tl }
        { \l__acro_alt_format_tl }
      {\LWR@textcurrentfont{#2}}% lwarp
    \group_end:
  }

\cs_gset_protected:Npn \acro_write_long:nn #1#2
```

```
{
  \mode_if_horizontal:F { \leavevmode }
  \group_begin:
    \bool_if:NTF \l__acro_custom_long_format_bool
      { \l__acro_custom_long_format_tl }
      { \use:n }
    {
      \use:x
        {
          \exp_not:n {#1}
          {
            \bool_if:NTF \l__acro_first_upper_bool
              { \exp_not:N \__acro_first_upper_case:n { \exp_not:n {
                  \LWR@textcurrentfont{#2}% lwarp
              } } }
              { \exp_not:n {\LWR@textcurrentfont{#2}} }% lwarp
          }
        }
    }
  \group_end:
}
}% before v2.10
}% before v3
\ExplSyntaxOff
```

File 13 **lwarp-acronym.sty**

§ 122 Package acronym

(*Emulates or patches code by* Tobias Oetiker.)

acronym (*Pkg*) acronym is patched for use by lwarp.

⚠ multiply-defined labels `\acresetall` does not work with cleveref, causing multiply-defined labels. lwarp patches acronym for HTML, but not for print mode.

for HTML output:

```
\LWR@ProvidesPackagePass{acronym}[2020/04/17]
```

Simplifies for HTML:

```
\expandafter\def\csname AC@\AC@prefix{}@acro\endcsname#1[#2]#3{%
  \ifAC@nolist%
  \else%
  \ifnum%
    \ifAC@printonlyused 1%
    \else\ifAC@printonlyreused 1%
    \else 0\fi\fi%
  =1\relax%
    \ifnum%
      \ifAC@printonlyused%
      \expandafter\ifx\csname acused@#1@once\endcsname\AC@used 1 \else 0 \fi%
      \else\ifAC@printonlyreused%
     \expandafter\ifx\csname acused@#1@twice\endcsname\AC@used 1 \else 0 \fi%
      \else 0 \fi\fi%
    =1\relax%
      \item[\protect\AC@hypertarget{#1}{%
        \AC@hyperref[acro:#1]{\aclabelfont{#2}\hfill}%
```

```
      }]\AC@hyperref[acro:#1]{#3}%
          \ifAC@withpage%
            \expandafter\ifx\csname r@acro:#1\endcsname\relax%
               \PackageInfo{acronym}{%
                 Acronym #1 used in text but not spelled out in
                 full in text}%
            \else%
%               \nobreak\leaders\hbox{%
%                   $\m@th\mkern\@dotsep mu\hbox{.}\mkern\@dotsep mu$%
%               }\hfill%
%               \nobreak\hb@xt@\@pnumwidth{%
%               \hfil\normalfont\normalcolor
                \qquad --- %    lwarp
                \AC@pageref{acro:#1}%
%               }%
            \fi%
          \fi\\%
    \fi%
  \else%
   \item[\protect\AC@hypertarget{#1}{\AC@hyperref[acro:#1]{\aclabelfont{#2}\hfill}}]%
        \AC@hyperref[acro:#1]{#3}%
  \fi%
  \fi%
  \begingroup
    \def\acroextra##1{}%
    \@bsphack
      \ifAC@printonlyreused%
        \protected@write\@auxout{}{%
          \string\newacro{#1}[%
            \expandafter\ifx\csname acused@#1@twice\endcsname\AC@used%
              \string\AC@hyperlink{#1}{#2}%
            \else%
              {#2}%
            \fi%
          ]{#3}%
        }%
      \else%
        \protected@write\@auxout{}{%
          \string\newacro{#1}[\string\AC@hyperlink{#1}{#2}]{#3}%
        }%
      \fi%
    \@esphack
  \endgroup
  \ignorespaces}
```

Uses `\textit` instead of `\itshape`:

```
\renewcommand{\acfia}[1]{%
  {\textit{\AC@acl{#1}}} (\ifAC@starred\acs*{#1}\else\acs{#1}\fi)}
```

Removes the mbox to allow math inside:

```
\renewcommand*\AC@acs[1]{%
%     \mbox{
 \expandafter\AC@get\csname fn@#1\endcsname\@firstoftwo{#1}}
% }
```

Fix for acronym labels in the captions of floats.

```
\renewcommand{\@starttoc}[1]{%
    \LWR@htmlelementclass{nav}{#1}
    \LetLtxMacro\@verridelabel\@gobble
    \LWR@orig@starttoc{#1}
    \LWR@htmlelementclassend{nav}{#1}
}
```

Modified for cleveref and lwarp:

```
\renewcommand*\AC@und@newl@bel[3]{%
    \@ifundefined{#1@#3}%
    {%
        \global\expandafter\let\csname#2@#3\endcsname\@nnil
        \global\expandafter\let\csname#2@#3@lwarp\endcsname\@nnil% lwarp
        \global\expandafter\let\csname#2@#3@cref\endcsname\@nnil% lwarp
    }%
    {%
        \global\expandafter\let\csname#1@#3\endcsname\relax
        \global\expandafter\let\csname#1@#3@lwarp\endcsname\relax% lwarp
        \global\expandafter\let\csname#1@#3@cref\endcsname\relax% lwarp
    }%
}%
```

Improve paragraph handling:

```
\BeforeBeginEnvironment{acronym}{\LWR@stoppars}
\AfterEndEnvironment{acronym}{\LWR@startpars}
```

Create hyperlinks, even though hyperref is only emulated:

```
\AtBeginDocument{
      \LetLtxMacro\AC@hyperlink\hyperlink
      \LetLtxMacro\AC@hyperref\hyperref
      \newcommand*\AC@raisedhypertarget[2]{%
%         \Hy@raisedlink{%
            \hypertarget{#1}{}%
%          }%
          #2}%
      \LetLtxMacro\AC@hypertarget\AC@raisedhypertarget
      \def\AC@phantomsection{%
%        \Hy@GlobalStepCount\Hy@linkcounter
%        \edef\@currentHref{section*.\the \Hy@linkcounter}%
%        \Hy@raisedlink{%
%          \hyper@anchorstart{\@currentHref}\hyper@anchorend
%        }%
%        \phantomsection%
      }%
}

\appto\LWR@restoreorigformatting{%
      \LetLtxMacro\AC@hyperlink\@secondoftwo%
      \LetLtxMacro\AC@hyperref\LWR@nullify@hyperref%
}
```

File 14 **lwarp-adjmulticol.sty**

§ 123 Package

adjmulticol

(*Emulates or patches code by* Boris Veytsman.)

adjmulticol (*Pkg*) adjmulticol is emulated.

Emulation similar to multicols is used, with adjusted margins. If the number of columns is specified as 1, it is set so, but if two or greater are used, lwarp allows a variable number of columns up to three.

for HTML output:

```
\LWR@ProvidesPackageDrop{adjmulticol}[2012/01/20]
\RequirePackage{multicol}
```

adjmulticols * {⟨*numcols*⟩} {⟨*left margi*⟩} {⟨*right margin*⟩}

```
\NewDocumentEnvironment{adjmulticols}{s m m m}
{%
```

Compute the margins, and limit to positive only:

```
\setlength{\LWR@templengthone}{#3}%
\ifdimcomp{\LWR@templengthone}{<}{0pt}{\setlength{\LWR@templengthone}{0pt}}{}%
\setlength{\LWR@templengthtwo}{#4}
\ifdimcomp{\LWR@templengthtwo}{<}{0pt}{\setlength{\LWR@templengthtwo}{0pt}}{}%
```

If one column is specified, use a of class singlecolumn, else use multicols:

```
\newcommand*{\LWR@mcolstype}{multicols}%
\ifnumcomp{#2}{=}{1}{\renewcommand*{\LWR@mcolstype}{singlecolumn}}{}%
```

Help avoid page overflow:

```
\LWR@forcenewpage%
```

Create the with the given margin and class:

```
\BlockClass[%
    \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
    \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}%
]{\LWR@mcolstype}%
}
{\endBlockClass}
```

File 15 **lwarp-addlines.sty**

§ 124 Package

addlines

(*Emulates or patches code by* Will Robertson.)

addlines (*Pkg*) addlines is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{addlines}[2018/12/05]

\newcommand\addlines{\@ifstar\addlines@a\addlines@a}
\newcommand\addlines@a[1][1]{}
\let\addline\addlines
\newcommand\removelines{\@ifstar\removelines@a\removelines@a}
\newcommand\removelines@a[1][1]{}
\let\removeline\removelines
\newcommand\squeezepage[1][0]{}
```

File 16 **lwarp-afterpage.sty**

§ 125 Package afterpage

(*Emulates or patches code by* David Carlisle.)

afterpage (*Pkg*) afterpage is emulated.

for HTML output: Discard all options for lwarp-afterpage:

```
\LWR@ProvidesPackageDrop{afterpage}[2014/10/28]

\newcommand{\afterpage}[1]{#1}
```

File 17 **lwarp-algorithm2e.sty**

§ 126 Package algorithm2e

(*Emulates or patches code by* Christophe Fiorio.)

algorithm2e (*Pkg*) algorithm2e is patched for use by lwarp.

For print output, captions are placed according to package options, but for HTML output captions are placed where used. Therefore, to have captions appear at the top of the algorithms for both print and HTML, place each captions at the top of each algorithm.

for HTML output:

```
\LWR@ProvidesPackagePass{algorithm2e}[2017/07/18]
```

For the list-of entries:

```
\renewcommand{\l@algocf}[2]{\hypertocfloat{1}{algocf}{loa}{#1}{#2}}
```

Select the lwarp float style according to the algorithm2e style:

```
\newcommand*{\LWR@floatstyle@algocf}{ruled}

\ifdefstring{\algocf@style}{boxed}{%
\renewcommand*{\LWR@floatstyle@algocf}{boxed}
}{}

```

```
\ifdefstring{\algocf@style}{boxruled}{%
\renewcommand*{\LWR@floatstyle@algocf}{boxruled}
}{}

\ifdefstring{\algocf@style}{plain}{%
\renewcommand*{\LWR@floatstyle@algocf}{plain}
}{}
```

Paragraph handling to allow line numbers under certain conditions:

```
\renewcommand{\algocf@everypar}{%
    \ifbool{LWR@algocf@dopars}{%
        \ifbool{LWR@doingstartpars}{%
            \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
                {}%
                {%
```

algorthm2e uses `\everypar`, so the open paragraph tag is generated here instead of `\LWR@openparagraph`:

```
                    \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%

                    \algocf@everyparnl\algocf@everyparhanging%
                }%
        }{}%
    }{}%
}
```

lwarp caption handling:

```
\renewcommand{\algocf@makecaption}[2]{%
    \LWR@HTML@caption@begin{algocf}%
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}
```

Print any caption where it is declared:

```
\renewcommand{\algocf@makecaption@plain}[2]{%
    \LWR@HTML@caption@begin{algocf}%
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}

\renewcommand{\algocf@makecaption@boxed}[2]{%
    \LWR@HTML@caption@begin{algocf}%
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}

\renewcommand{\algocf@makecaption@ruled}[2]{%
    \LWR@HTML@caption@begin{algocf}%
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}
```

Turn off line numbering while making the caption:

```
\long\def\algocf@latexcaption#1[#2]#3{% original definition of caption
\boolfalse{LWR@algocf@dopars}%      lwarp
  \par%
  \addcontentsline{\csname ext@#1\endcsname}{#1}%
 {\protect\numberline{\csname the#1\endcsname}{\ignorespaces \LWR@isolate{#2}}}%
  \begingroup%
  \@parboxrestore%
  \if@minipage%
    \@setminipage%
  \fi%
  \normalsize%
  \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par%
  \endgroup%
\booltrue{LWR@algocf@dopars}%       lwarp
}
```

Line numbers are printed in a of class `alg2elinenumber`:

```
\renewcommand{\algocf@printnl}[1]{%
    \InlineClass{alg2elinenumber}{\NlSty{#1}}~%
}%
```

While initializing an algorithm environment, locally declare the style of a regular figure to be the same as the algorithm style, in case the `figure` option was used.

```
\preto\@algocf@init{%
  \edef\LWR@floatstyle@figure{\LWR@floatstyle@algocf}%
}
```

For lwarp, the algorithm is not assembled inside a box, since `latexi`mages would not work, so the captions are printed where declared.

```
\renewcommand{\@algocf@start}{%
   \let\@mathsemicolon=\;\def\;{\ifmmode\@mathsemicolon\else\@endalgoln\fi}%
%     \raggedright%
    \AlFnt{}%
    \booltrue{LWR@algocf@dopars}% lwarp
}

\renewcommand{\@algocf@finish}{%
    \boolfalse{LWR@algocf@dopars}% lwarp
    \lineskip\normallineskip\setlength{\skiptotal}{\@defaultskiptotal}%
    \let\;=\@mathsemicolon%
    \let\]=\@emathdisplay%
}
```

Use an HTML break:

```
\renewcommand{\BlankLine}{%
\LWR@stoppars%
\LWR@htmltagc{br /}%
\LWR@startpars%
}
```

Simplified for HTML. The paragraph handling must be preserved.

```
\renewcommand{\SetKwInOut}[2]{%
  \algocf@newcommand{#1}[1]{%
    \ifthenelse{\boolean{algocf@hanginginout}}%
```

```
        {\relax}%
        {\algocf@seteveryparhanging{\relax}}%
    \ifthenelse{\boolean{algocf@inoutnumbered}}%
        {\relax}%
        {\algocf@seteveryparnl{\relax}}%
    {%
            \KwSty{#2\algocf@typo:}%
        ~##1\par%
    }%
    \algocf@linesnumbered% reset the numbering of the lines
    \ifthenelse{\boolean{algocf@hanginginout}}%
        {\relax}%
        {\algocf@reseteveryparhanging}%
  }%
}%

\renewcommand{\ResetInOut}[1]{}%
```

Each of the following creates a of a given class, and turns off line numbering while creating the tags:

```
\renewcommand{\algocf@Vline}[1]{%
    \boolfalse{LWR@algocf@dopars}%
    \begin{BlockClass}{alg2evline}
    \booltrue{LWR@algocf@dopars}%
    #1
    \boolfalse{LWR@algocf@dopars}%
    \end{BlockClass}
    \booltrue{LWR@algocf@dopars}%
}

\renewcommand{\algocf@Vsline}[1]{%
    \boolfalse{LWR@algocf@dopars}%
    \begin{BlockClass}{alg2evsline}
    \booltrue{LWR@algocf@dopars}%
    #1
    \boolfalse{LWR@algocf@dopars}%
    \end{BlockClass}
    \booltrue{LWR@algocf@dopars}%
}

\renewcommand{\algocf@Noline}[1]{%
    \boolfalse{LWR@algocf@dopars}%
    \begin{BlockClass}{alg2enoline}
    \booltrue{LWR@algocf@dopars}%
    #1
    \boolfalse{LWR@algocf@dopars}%
    \end{BlockClass}
    \booltrue{LWR@algocf@dopars}%
}
```

The [H] environment is converted to a regular float, which in HTML is placed where declared. Reusing the regular float allows the [H] version to reuse the ruled and boxed options.

```
\LetLtxMacro\algocf@Here\algocf
\LetLtxMacro\endalgocf@Here\endalgocf
```

File 18 **lwarp-algorithmicx.sty**

§ 127 Package algorithmicx

(*Emulates or patches code by* Szász János.)

`algorithmicx` (*Pkg*) algorithmicx is supported with minor adjustments.

for HTML output:

```
\LWR@ProvidesPackagePass{algorithmicx}[2005/04/27]
```

Inside the `algorithmic` environment, level indenting is converted to a of the required length, and comments are placed inside a which is floated right.

⚠ package conflicts If using `\newfloat`, trivfloat, and/or algorithmicx together, see section 639.1.

```
\AtBeginEnvironment{algorithmic}{%
%
\let\origALG@doentity\ALG@doentity%
%
\renewcommand*{\ALG@doentity}{%
\origALG@doentity%
\LWR@htmltagc{%
    span style=\textquotedbl{}%
        width:\LWR@printlength{\ALG@thistlm}; display:inline-block;%
    \textquotedbl%
}%
\ifbool{FormatWP}{%
\setlength{\LWR@templengthone}{\the\ALG@thistlm}%
\whiledo{\lengthtest{\LWR@templengthone>1em}}{%
\quad%
\addtolength{\LWR@templengthone}{-1em}%
}%
}{}%
\LWR@htmltagc{/span}%
}%

\let\LWR@origComment\Comment%

\renewcommand{\Comment}[1]{%
    \InlineClass{floatright}{\LWR@origComment{#1}}%
}%
}

\renewcommand\algorithmiccomment[1]{%
\hfill\HTMLunicode{25B7} #1% white right triangle
}%
```

File 19 **lwarp-alltt.sty**

§ 128 Package alltt

(*Emulates or patches code by* Johannes Braams.)

alltt (*Pkg*) alltt is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{alltt}[1997/06/16]

\AfterEndPreamble{
\LWR@traceinfo{Patching alltt.}

\AtBeginEnvironment{alltt}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
            \LWR@forcenewpage
```

Vertical spacing changes if inside a list.

```
            \LWR@atbeginverbatim{alltt}%
        }%
}

\AfterEndEnvironment{alltt}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
```

Vertical spacing changes if inside a list.

```
            \LWR@afterendverbatim%
        }%
}


}
```

File 20 **lwarp-amscdx.sty**

§ 129 Package

amscdx

(*Emulates or patches code by* Martin Vermeer.)

amscdx (*Pkg*) amscdx is used as-is for SVG math.

⚠ MathJax For MathJax, a warning notes that the CD environment must be enclosed between \displaymathother and \displaymathnormal.

for HTML output:

```
\LWR@ProvidesPackagePass{amscdx}[2019/07/02]

\begin{warpMathJax}
\CustomizeMathJax{%
    \renewenvironment{CD}
     {\text{(Use \unicode{x005C}displaymathother before the CD enviroment.) \quad}}
     {\quad \text{(Use \unicode{x005C}displaymathnormal after the CD enviroment.)}}
}

\CustomizeMathJax{\newcommand{\CDfattrue}{}}
\CustomizeMathJax{\newcommand{\CDfatfalse}{}}
\CustomizeMathJax{\newcommand{\CDashtrue}{}}
\CustomizeMathJax{\newcommand{\CDashfalse}{}}
```

```
\CustomizeMathJax{\newcommand{\CDlor}[1]{}}
\end{warpMathJax}
```

File 21 **lwarp-amsmath.sty**

§ 130 Package

amsmath

(*Emulates or patches code by* American Mathematical Society, LaTeX3 Project.)

amsmath (*Pkg*) amsmath is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{amsmath}[2017/09/02]
```

\dotso An HTML text-mode version.

```
\newcommand*{\LWR@HTML@dotso}{\textellipsis\ }
\LWR@formatted{dotso}
```

Patches to allow \eqref inside a caption:

```
\def\maketag@@@#1{\text{#1}}
\def\tagform@#1{\maketag@@@{(\ignorespaces#1\unskip)}}
```

Patches for $\mathcal{AMS}$ math \tag macro to remember the first tag:

```
\ifbool{mathjax}{}{% not mathjax

\LetLtxMacro\LWR@origmake@df@tag@@\make@df@tag@@
\LetLtxMacro\LWR@origmake@df@tag@@@\make@df@tag@@@

\renewcommand*{\make@df@tag@@}[1]{%
    \LWR@remembertag{#1}%
    \LWR@origmake@df@tag@@{#1}%
}

\renewcommand*{\make@df@tag@@@}[1]{%
    \LWR@remembertag{#1}%
    \LWR@origmake@df@tag@@@{#1}%
}

}% not mathjax
```

For nesting $\mathcal{AMS}$ environments:

```
\newcounter{LWR@amsmathdepth}
\setcounter{LWR@amsmathdepth}{0}
```

The following $\mathcal{AMS}$ environments are patched in-place:

LWR@maxfields@ (*Ctr*) A copy of maxfields@ as it was passed. This is used to generate the mandatory argument for alignat and alignat* when using MathJax.

```
\newcounter{LWR@maxfields@}

\xpatchcmd{\start@align}
```

```
    {\maxfields@#3\relax}
    {%
        \maxfields@#3\relax%
        \setcounter{LWR@maxfields@}{#3}%
    }
    {}
    {\LWR@patcherror{amsmath}{start@align}}
```

`\LWR@amsmathenv@@before` * {⟨*environment name*⟩}

`*` if the environment was starred.

Embeds the environment inside a `lateximage`.

```
\NewDocumentCommand{\LWR@amsmathenv@@before}{s m}{%
    \IfBooleanTF{#1}{
        \begin{BlockClass}{displaymath}
    }{
        \begin{BlockClass}{displaymathnumbered}
    }
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{#2}]*%
    \LWR@applyxfakebold%
}
```

`\LWR@amsmathenv@before` * {⟨*environment name*⟩}

`*` if the environment was starred.

Embeds the environment with MathJax or a `lateximage`.

```
\NewDocumentCommand{\LWR@amsmathenv@before}{s m}{%
    \ifnumequal{\value{LWR@amsmathdepth}}{0}{%
        \LWR@stoppars%
      \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
        {
            \LWR@syncmathjax
            \boolfalse{LWR@amsmultline}
            \ifstrequal{#2}{multline}{\booltrue{LWR@amsmultline}}{}
            \ifstrequal{#2}{multline*}{\booltrue{LWR@amsmultline}}{}
```

⚠ autonum's "+" environments are not supported by MathJax.

```
            \LWR@beginhideamsmath
        }
        {
            \IfBooleanTF{#1}{
                \LWR@amsmathenv@@before*{#2}
            }{
                \LWR@amsmathenv@@before{#2}
            }
        }
    }{}
    \addtocounter{LWR@amsmathdepth}{1}
}
```

`\LWR@amsmathenv@@after`

Embeds the environment inside a `lateximage`.

```
\newcommand*{\LWR@amsmathenv@@after}{%
    \end{lateximage}\end{BlockClass}\LWR@startpars%
```

```
}
```

`\LWR@amsmathenv@after` * {⟨*environment name*⟩}

`*` if the environment was starred. Ignored here, only used for a consistent syntax.

Embeds the environment with MathJax or a `lateximage`.

```
\NewDocumentCommand{\LWR@amsmathenv@after}{s m}{%
    \ifnumequal{\value{LWR@amsmathdepth}}{1}{%
      \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
        {
            \LWR@endhideamsmath
            \boolfalse{LWR@amsmultline}
            \LWR@addmathjax{#2}{\the\@envbody}
        }
        {\LWR@amsmathenv@@after}
```

Clear the single-use `alt` text:

```
        \gdef\LWR@ThisAltText{}%
    }{}
    \addtocounter{LWR@amsmathdepth}{-1}
}
```

`multline` (*env.*)

```
\BeforeBeginEnvironment{multline}{\LWR@amsmathenv@before{multline}}

\AfterEndEnvironment{multline}{\LWR@amsmathenv@after{multline}}
```

`multline*` (*env.*)

```
\BeforeBeginEnvironment{multline*}{\LWR@amsmathenv@before*{multline*}}

\AfterEndEnvironment{multline*}{\LWR@amsmathenv@after*{multline*}}

```

`gather` (*env.*)

```
\BeforeBeginEnvironment{gather}{\LWR@amsmathenv@before{gather}}

\AfterEndEnvironment{gather}{\LWR@amsmathenv@after{gather}}
```

`gather*` (*env.*)

```
\BeforeBeginEnvironment{gather*}{\LWR@amsmathenv@before*{gather*}}

\AfterEndEnvironment{gather*}{\LWR@amsmathenv@after*{gather*}}
```

`align` (*env.*)

```
\BeforeBeginEnvironment{align}{\LWR@amsmathenv@before{align}}

\AfterEndEnvironment{align}{\LWR@amsmathenv@after{align}}
```

`align*` (*env.*)

```
\BeforeBeginEnvironment{align*}{\LWR@amsmathenv@before*{align*}}

\AfterEndEnvironment{align*}{\LWR@amsmathenv@after*{align*}}
```

`flalign` (*env.*)

```
\BeforeBeginEnvironment{flalign}{\LWR@amsmathenv@before{flalign}}

\AfterEndEnvironment{flalign}{\LWR@amsmathenv@after{flalign}}
```

`flalign*` (*env.*)

```
\BeforeBeginEnvironment{flalign*}{\LWR@amsmathenv@before*{flalign*}}

\AfterEndEnvironment{flalign*}{\LWR@amsmathenv@after*{flalign*}}
```

`alignat` (*env.*)

```
\BeforeBeginEnvironment{alignat}{\LWR@amsmathenv@before{alignat}}

\AfterEndEnvironment{alignat}{\LWR@amsmathenv@after{alignat}}
```

`alignat*` (*env.*)

```
\BeforeBeginEnvironment{alignat*}{\LWR@amsmathenv@before*{alignat*}}

\AfterEndEnvironment{alignat*}{\LWR@amsmathenv@after*{alignat*}}
```

```
\AtBeginEnvironment{subequations}{
    \renewcommand*{\theMathJaxsubequations}{1}
    \renewcommand*{\theMathJaxsection}{\theparentequation}
    \renewcommand*{\theMathJaxequation}{\arabic{equation}}
}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\intertext}[1]{\text{#1}\notag \\}}
\CustomizeMathJax{\let\Hat\hat}
\CustomizeMathJax{\let\Check\check}
\CustomizeMathJax{\let\Tilde\tilde}
\CustomizeMathJax{\let\Acute\acute}
\CustomizeMathJax{\let\Grave\grave}
\CustomizeMathJax{\let\Dot\dot}
\CustomizeMathJax{\let\Ddot\ddot}
\CustomizeMathJax{\let\Breve\breve}
\CustomizeMathJax{\let\Bar\bar}
\CustomizeMathJax{\let\Vec\vec}
\end{warpMathJax}
```

File 22 **lwarp-amsthm.sty**

§ 131 Package amsthm

(*Emulates or patches code by* Publications Technical Group — American Mathematical Society.)

The original source code is located in amsclass.dtx, and printed in amsclass.pdf.

amsthm (*Pkg*) amsthm is patched for use by lwarp.

Table 19: amsthm package — css styling of theorems and proofs

Theorem: of class amsthmbody<theoremstyle>

Theorem Name: of class amsthmname<theoremtyle>

Theorem Number: of class amsthmnumber<theoremstyle>

Theorem Note: of class amsthmnote<theoremstyle>

Proof: of class amsthmproof

Proof Name: of class amsthmproofname

where <theoremstyle> is plain, definition, etc.

for HTML output: amsthm must be loaded before mdframed:

```
\IfPackageLoadedTF{mdframed}{
    \PackageError{lwarp}
    {%
        Package mdframed must be loaded after package amsthm.\MessageBreak
        Enter 'H' for solutions%
    }
    {%
        Move ``\protect\usepackage{amsthm}'' before
        ``\protect\usepackage{mdframed}''.\MessageBreak
        Package amsthm may be loaded by something else,\MessageBreak
        which must also be moved before mdframed.%
    }
}
{\relax}
```

Necessary for \text, used by \openbox, etc., below:

```
\RequirePackage{amsmath}
```

```
\LWR@ProvidesPackagePass{amsthm}[2017/10/31]
```

Storage for the style being used for new theorems:

```
\newcommand{\LWR@newtheoremstyle}{plain}
```

Patched to remember the style being used for new theorems:

```
\renewcommand{\theoremstyle}[1]{%
  \@ifundefined{th@#1}{%
    \PackageWarning{amsthm}{Unknown theoremstyle `#1'}%
    \thm@style{plain}%
    \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
  }{%
    \thm@style{#1}%
    \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
  }%
}
```

Patched to remember the style for this theorem type:

```
\def\@xnthm#1#2{%
  \csedef{LWR@thmstyle#2}{\LWR@newtheoremstyle}% lwarp
  \let\@tempa\relax
  \@xp\@ifdefinable\csname #2\endcsname{%
    \global\@xp\let\csname end#2\endcsname\@endtheorem
    \ifx *#1% unnumbered, need to get one more mandatory arg
      \edef\@tempa##1{%
        \gdef\@xp\@nx\csname#2\endcsname{%
          \@nx\@thm{\@xp\@nx\csname th@\the\thm@style\endcsname}%
            {}{##1}}}%
    \else % numbered theorem, need to check for optional arg
      \def\@tempa{\@oparg{\@ynthm{#2}}[]}%
    \fi
    \AtBeginEnvironment{#2}{%                                  lwarp
        \edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#2}}%  lwarp
    }%                                                         lwarp
  }%
  \@tempa%
}
```

Patched to enclose with css:

```
\newcommand{\LWR@haveamsthmname}{
    \renewcommand{\thmname}[1]{%
        \InlineClass{amsthmname\LWR@thisthmstyle}{##1}%
    }
}

\newcommand{\LWR@haveamsthmnumber}{
    \renewcommand{\thmnumber}[1]{%
        \InlineClass{amsthmnumber\LWR@thisthmstyle}{##1}%
    }
}

\newcommand{\LWR@haveamsthmnote}{
    \renewcommand{\thmnote}[1]{%
        \InlineClass{amsthmnote\LWR@thisthmstyle}{##1}%
    }
}

\LWR@haveamsthmname
\LWR@haveamsthmnumber
\LWR@haveamsthmnote
```

Patched for css:

```
\def\@begintheorem#1#2[#3]{%

    \GetTitleString{#3}%                                lwarp
    \let\@currentlabelname\GetTitleStringResult%        lwarp
    \item[%

    \LWR@newautopagelabel{page}\LWR@orignewline%

%   \deferred@thm@head{
%     \the\thm@headfont \thm@indent
```

```
    \@ifempty{#1}{\let\thmname\@gobble}{\LWR@haveamsthmname}%       lwarp
    \@ifempty{#2}{\let\thmnumber\@gobble}{\LWR@haveamsthmnumber}%   lwarp
    \@ifempty{#3}{\let\thmnote\@gobble}{\LWR@haveamsthmnote}%       lwarp
    \thm@swap\swappedhead\thmhead{#1}{#2}{#3}%
    \the\thm@headpunct % space
    \thmheadnl % possibly a newline.
    \hskip\thm@headsep
%   }%
    ]%
  \ignorespaces}
```

Patched for css:

```
\def\@thm#1#2#3{%
  \ifhmode\unskip\unskip\par\fi
  \normalfont
  \LWR@forcenewpage%                              lwarp

  \LWR@printpendingfootnotes%                     lwarp

  \BlockClass{amsthmbody\LWR@thisthmstyle}%       lwarp
  \trivlist
  \let\thmheadnl\relax
  \let\thm@swap\@gobble
  \thm@notefont{\fontseries\mddefault\upshape}%
  \thm@headpunct{.}% add period after heading
  \thm@headsep 5\p@ plus\p@ minus\p@\relax
  \thm@space@setup
  #1% style overrides
  \@topsep \thm@preskip               % used by thm head
  \@topsepadd \thm@postskip           % used by \@endparenv
  \def\@tempa{#2}\ifx\@empty\@tempa
    \def\@tempa{\@oparg{\@begintheorem{#3}{}}[]}%
  \else
    \refstepcounter{#2}%
    \def\@tempa{\@oparg{\@begintheorem{#3}{\csname the#2\endcsname}}[]}%
  \fi
  \@tempa%
}
```

cleveref patches `\@thm` to do `\cref@thmoptarg` if an optional argument is given. lwarp then patches `\cref@thmoptarg` `\AtBeginDocument`.

```
\AtBeginDocument{%
\def\cref@thmoptarg[#1]#2#3#4{%
    \ifhmode\unskip\unskip\par\fi%
    \normalfont%
    \LWR@forcenewpage%                            lwarp

    \LWR@printpendingfootnotes%                     lwarp

    \BlockClass{amsthmbody\LWR@thisthmstyle}%     lwarp
    \trivlist%
    \let\thmheadnl\relax%
    \let\thm@swap\@gobble%
    \thm@notefont{\fontseries\mddefault\upshape}%
    \thm@headpunct{.}% add period after heading
    \thm@headsep 5\p@ plus\p@ minus\p@\relax%
```

```
    \thm@space@setup%
    #2% style overrides
    \@topsep \thm@preskip               % used by thm head
    \@topsepadd \thm@postskip           % used by \@endparenv
    \def\@tempa{#3}\ifx\@empty\@tempa%
        \def\@tempa{\@oparg{\@begintheorem{#4}{}}[]}%
    \else%
        \refstepcounter[#1]{#3}%  <<< cleveref modification
       \def\@tempa{\@oparg{\@begintheorem{#4}{\csname the#3\endcsname}}[]}%
    \fi%
    \@tempa
}%
}% AtBeginDocument

\def\@endtheorem{%
    \endtrivlist%

  \LWR@printpendingfootnotes%                   lwarp

    \endBlockClass%
    \@endpefalse%
}
```

Proof QED symbol:

```
\AtBeginDocument{
\@ifundefined{LWR@orig@openbox}{
\LetLtxMacro\LWR@orig@openbox\openbox
\LetLtxMacro\LWR@orig@blacksquare\blacksquare
\LetLtxMacro\LWR@orig@Box\Box

\def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
\def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
\def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box

\appto\LWR@restoreorigformatting{%
    \LetLtxMacro\openbox\LWR@orig@openbox%
    \LetLtxMacro\blacksquare\LWR@orig@blacksquare%
    \LetLtxMacro\Box\LWR@orig@Box%
}% appto
}{}% @ifundefined
}% AtBeginDocument
```

Patched to add a :

```
\DeclareRobustCommand{\qed}{%
  \ifmmode \mathqed
  \else
%     \leavevmode\unskip\penalty9999 \hbox{}\nobreak\hfill
%     \quad\hbox{\qedsymbol}%
        \InlineClass{theoremendmark}{\qedsymbol}%   lwarp
  \fi
}
```

Patched for css:

```
\renewenvironment{proof}[1][\proofname]{\par
  \LWR@forcenewpage% lwarp
```

```
\LWR@printpendingfootnotes%                    lwarp

  \BlockClass{amsthmproof}% lwarp
  \LWR@newautopagelabel{page}%
\pushQED{\qed}%
\normalfont \topsep6\p@\@plus6\p@\relax
\trivlist
\item[
     \InlineClass{amsthmproofname}{#1\@addpunct{.}}]\ignorespaces% changes
}{%
\popQED\endtrivlist%

\LWR@printpendingfootnotes%                    lwarp

\endBlockClass% lwarp
\@endpefalse
}
```

File 23 **lwarp-anonchap.sty**

§ 132 Package anonchap

(*Emulates or patches code by* Peter Wilson.)

anonchap (*Pkg*) anonchap is emulated.

tocloft (*Pkg*) If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its `titles` option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

⚠ tocloft & other packages

The code is shared by tocbibind.

for HTML output:

```
\LWR@ProvidesPackageDrop{anonchap}[2009/08/03]

\newcommand{\simplechapter}[1][\@empty]{%
    \def\@chapcntformat##1{%
        #1~\csname the##1\endcsname\simplechapterdelim\quad%
    }%
}

\newcommand{\restorechapter}{%
\let\@chapcntformat\@seccntformat%
}
```

File 24 **lwarp-anysize.sty**

§ 133 Package anysize

(*Emulates or patches code by* Michael Salzenberg, Thomas Esser.)

anysize (*Pkg*) anysize is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{anysize}[1994/08/13]
```

```
\def\papersize#1#2{}
\def\marginsize#1#2#3#4{}
```

File 25 **lwarp-appendix.sty**

§ 134 Package appendix

(*Emulates or patches code by* Peter Wilson.)

appendix (*Pkg*) appendix is patched for use by lwarp.

⚠ incorrect TOC link During HTML conversion, the option toc without the option page results in a TOC link to whichever section was before the appendices environment. It is recommended to use both toc and also page at the same time.

for HTML output:

```
\LWR@ProvidesPackagePass{appendix}[2009/09/02]

\renewcommand*{\@chap@pppage}{%
\part*{\appendixpagename}
 \if@dotoc@pp
 \addappheadtotoc
 \fi
}

\renewcommand*{\@sec@pppage}{%
\part*{\appendixpagename}
 \if@dotoc@pp
 \addappheadtotoc
 \fi
}
```

File 26 **lwarp-ar.sty**

§ 135 Package ar

(*Emulates or patches code by* Agostino De Marco.)

ar (*Pkg*) ar is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{ar}[2012/01/23]
```

Measure and print the width of the supplied glyph.

```
\newlength{\LWR@ar@width}

\newcommand*{\LWR@ar@printwidth}[1]{%
    \setlength{\LWR@ar@width}{\widthof{#1}}%
    width:%
    \LWR@convertto{em}{\the\LWR@ar@width}em%
}
```

The HTML version of \AR:

```
\newrobustcmd*{\LWR@HTML@AR}{%
```

Start a hashed `lateximage`, additionally hashed by the font series, with a width depending on the given glyph:

```
  \begin{lateximage}*[AR][\LWR@f@series][\LWR@ar@printwidth{\LWR@print@AR}]%
```

For text mode, set the font series according to the HTML font series:

```
    \ifmmode\else\csuse{LWR@orig\LWR@f@series series}\fi%
```

Print the original glyph using the newly set font series:

```
    \LWR@print@AR%
```

Done.

```
    \end{lateximage}%
}
```

Combine the print and HTML versions:

```
\LWR@formatted{AR}

\newrobustcmd*{\LWR@HTML@ARb}{%
    \begin{lateximage}*[AR][b][\LWR@ar@printwidth{\LWR@print@ARb}]%
    \LWR@print@ARb%
    \end{lateximage}%
}
\LWR@formatted{ARb}

\newrobustcmd*{\LWR@HTML@ARss}{%
  \begin{lateximage}*[ARss][\LWR@f@series][\LWR@ar@printwidth{\LWR@print@ARss}]%
    \ifmmode\else\csuse{LWR@orig\LWR@f@series series}\fi%
    \LWR@print@ARss%
    \end{lateximage}%
}
\LWR@formatted{ARss}

\newrobustcmd*{\LWR@HTML@ARssb}{%
    \begin{lateximage}*[AR][ssb][\LWR@ar@printwidth{\LWR@print@ARssb}]%
    \LWR@print@ARssb%
    \end{lateximage}%
}
\LWR@formatted{ARssb}

\newrobustcmd*{\LWR@HTML@ARtt}{%
    \begin{lateximage}*[AR][tt][\LWR@ar@printwidth{\LWR@print@ARtt}]%
    \LWR@print@ARtt%
    \end{lateximage}%
}
\LWR@formatted{ARtt}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\AR}{\mathit{A\!\!R}}}
\CustomizeMathJax{\newcommand{\ARb}{\boldsymbol{A\!\!R}}}
\end{warpMathJax}
```

File 27 **lwarp-arabicfront.sty**

§ 136 Package arabicfront

arabicfront (*Pkg*) arabicfront is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{arabicfront}[2006/09/03]
```

File 28 **lwarp-array.sty**

§ 137 Package array

array (*Pkg*) array is used as-is for print output, and emulated for HTML.

plarray and plextarray do not affect `\firsthline` or `\lasthline`, and so are not affected by the following.

for HTML output: If array is not yet loaded, remove the default nullfied macros:

```
\IfPackageLoadedTF{array}{}{%
    \let\firsthline\relax
    \let\lasthline\relax
}

\LWR@ProvidesPackagePass{array}[2018/12/30]
```

Provide simplified column types for HTML:

```
\HTMLnewcolumntype{w}[2]{#1}
\HTMLnewcolumntype{W}[2]{#1}
```

More HTML versions:

```
\newcommand*{\LWR@HTML@firsthline}{\LWR@HTMLhline}%
\LWR@expandableformatted{firsthline}

\newcommand*{\LWR@HTML@lasthline}{\LWR@HTMLhline}%
\LWR@expandableformatted{lasthline}


\let\tabularnewline\\
\providecommand*{\LWR@HTML@tabularnewline}{\LWR@tabularendofline}
\LWR@formatted{tabularnewline}
```

For MathJax:

```
\CustomizeMathJax{
    \newcommand{\multicolumn}[3]{#3}% only uses one cell
}
```

File 29 **lwarp-arydshln.sty**

§ 138 Package

arydshln

(*Emulates or patches code by* Hiroshi Nakashima.)

arydshln (*Pkg*) arydshln heavily patches tabular code, so the actual package is not used. arydshln is emulated for HTML tabular, and reverts to solid rules for SVG math array and tabular in a lateximage.

CSS is not able to display a double-dashed border, so a single-dashed rule is displayed as a single-dashed border, and a double-dashed rule is displayed as a thicker single-dashed border.

For MathJax, limited emulation is provided for math mode.

for HTML output: array is required to allow \newcolumn below.

```
\RequirePackage{array}

\LWR@ProvidesPackageDrop{arydshln}[2018/09/26]
```

Ignored, but included for source compatibility:

```
\newdimen\dashlinedash \dashlinedash4pt %
\newdimen\dashlinegap \dashlinegap4pt %
\let\hdashlinewidth\dashlinedash
\let\hdashlinegap\dashlinegap

\def\ADLnullwide{}
\def\ADLsomewide{}
\def\ADLnullwidehline{}
\def\ADLsomewidehline{}

\def\ADLactivate{}
\def\ADLinactivate{}
\newcommand*{\ADLdrawingmode}[1]{}
\newcommand*{\ADLnoshorthanded}{}
\newcommand*{\dashgapcolor}[2][]{}
\newcommand*{\nodashgapcolor}{}
```

In a lateximage, revert to solid vertical rules:

```
\appto\LWR@restoreorigformatting{%
\newcolumntype{:}{|}%
\newcolumntype{;}[1]{|}%
\LetLtxMacro\hdashline\hline%
}
```

Some of these macros are already defined as temporary placeholders in the lwarp core, so they must be redefined here.

The emulated defaults also work for an emulated print mode inside a lateximage:

```
\def\hdashline{
```

```
%     \adl@hdashline\adl@ihdashline
    \adl@hdashline\adl@inactivehdl
}
\def\adl@hdashline#1{\noalign{\ifnum0=`}\fi
%         \ifadl@zwhrule \vskip-\arrayrulewidth
%         \else
%             \adl@hline\adl@connect\arrayrulewidth
                \hrule \@height \arrayrulewidth% lwarp
%         \fi
        \@ifnextchar[%]
                    {#1}%
                    {#1[%
%                         \dashlinedash/\dashlinegap
                       1pt/1pt
                    ]}}
% \def\adl@ihdashline[#1/#2]{\ifnum0=`{\fi}%
%         \multispan{\adl@columns}\unskip \adl@hcline\z@[#1/#2]%
%         \noalign{\ifnum0=`}\fi
%         \futurelet\@tempa\adl@xhline}
\def\adl@inactivehdl[#1/#2]{
%         \ifadl@zwhrule \vskip-\arrayrulewidth \fi
        \hrule\@height\arrayrulewidth
        \futurelet\@tempa\adl@xhline}
\def\adl@xhline{\ifx\@tempa\hline \adl@ixhline\fi
        \ifx\@tempa\hdashline \adl@ixhline\fi
        \ifnum0=`{\fi}}
\def\adl@ixhline{\vskip\doublerulesep \adl@hline\relax\doublerulesep}
\def\adl@hline#1#2{%
% \@tempcnta#2
%         \global\advance\adl@totalheight\@tempcnta
%         \xdef\adl@rowsL{\adl@rowsL
%                 (#1/\number\@tempcnta);}%
%         \xdef\adl@rowsR{\adl@rowsR
%                 (#1/\number\@tempcnta);}
}

\def\cdashline#1{\noalign{\ifnum0=`}\fi
        \@ifnextchar[%]
%                     {\adl@cdline[#1]}%
%                     {\adl@cdline[#1][\dashlinedash/\dashlinegap]}
                    {\adl@inactivecdl[#1]}%
                    {\adl@inactivecdl[#1][\dashlinedash/\dashlinegap]}
}

\def\adl@inactivecdl[#1-#2][#3]{\ifnum0=`{\fi}\cline{#1-#2}}


\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\firsthdashline}[1][]{\hdashline}}
\CustomizeMathJax{\let\lasthdashline\firsthdashline}
\CustomizeMathJax{\let\cdashline\cline}
\end{warpMathJax}
```

File 30 **lwarp-asymptote.sty**

§ 139 Package asymptote

(*Emulates or patches code by* Andy Hammerlindl, John Bowman, Tom Prince.)

asymptote (*Pkg*) asymptote is patched for use by lwarp.

To compile:

```
pdflatex project.tex
asy project-*.asy
pdflatex project.tex

lwarpmk print
asy project-*.asy
lwarpmk print1
lwarpmk print1

lwarpmk html
asy project_html-*.asy
lwarpmk html1
lwarpmk html1
lwarpmk limages
```

for HTML output:

```
\LWR@ProvidesPackagePass{asymptote}[2016/11/26]

\BeforeBeginEnvironment{asy}{%
    \begin{lateximage}[-asymptote-~\PackageDiagramAltText]%
}
\AfterEndEnvironment{asy}{\end{lateximage}}

\xpatchcmd{\asyinclude}
    {\begingroup}
    {\begin{lateximage}[-asymptote-~\PackageDiagramAltText]}
    {}
    {\LWR@patcherror{asymptote}{asyinclude-begingroup}}

\xpatchcmd{\asyinclude}
    {\endgroup}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{asymptote}{asyinclude-endgroup}}
```

File 31 **lwarp-atbegshi.sty**

§ 140 Package atbegshi

(*Emulates or patches code by* Heiko Oberdiek.)

atbegshi (*Pkg*) atbegshi is ignored.

for HTML output: Discard all options for lwarp-atbegshi:

```
\LWR@ProvidesPackageDrop{atbegshi}[2011/10/05]

\let\AtBeginShipout\relax
\let\AtBeginShipoutNext\relax
\let\AtBeginShipoutFirst\relax
\let\AtBeginShipoutDiscard\relax
\let\AtBeginShipoutInit\relax
\let\AtBeginShipoutAddToBox\relax
```

```
\let\AtBeginShipoutAddToBoxForeground\relax
\let\AtBeginShipoutUpperLeft\relax
\let\AtBeginShipoutUpperLeftForeground\relax
\let\AtBeginShipoutOriginalShipout\relax

\newcommand*{\AtBeginShipout}[1]{}
\newbox\AtBeginShipoutBox
\newcommand*{\AtBeginShipoutNext}[1]{}
\newcommand*{\AtBeginShipoutFirst}[1]{}
\newcommand*{\AtBeginShipoutDiscard}{}
\newcommand*{\AtBeginShipoutInit}{}
\newcommand*{\AtBeginShipoutAddToBox}[1]{}
\newcommand*{\AtBeginShipoutAddToBoxForeground}[1]{}
\newcommand*{\AtBeginShipoutUpperLeft}[1]{}
\newcommand*{\AtBeginShipoutUpperLeftForeground}[1]{}
\newcommand*{\AtBeginShipoutOriginalShipout}[1]{}
\def\AtBeginShipoutBoxWidth{0pt}
\def\AtBeginShipoutBoxHeight{0pt}
\def\AtBeginShipoutBoxDepth{0pt}
```

File 32 **lwarp-attachfile.sty**

§ 141 Package

attachfile

(*Emulates or patches code by* Scott Pakin.)

attachfile (*Pkg*) attachfile is patched for use by lwarp.

⚠ Metadata is ignored for now.

for HTML output:

```
\LWR@ProvidesPackagePass{attachfile}[2016/09/18]
```

Encloses each icon:

```
\newenvironment*{LWR@attachfile@icon}
{
    \begin{lateximage}*%
        [-attachfile-]%
        [%
            \detokenize\expandafter{\atfi@icon@icon}-%
            \detokenize\expandafter{\atfi@color@rgb}%
        ]%
}
{
    \end{lateximage}
}
```

Each icon is enclosed inside a `LWR@attachfile@icon` environment:

```
\xpretocmd{\atfi@acroGraph}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroGraph}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPaperclip}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPaperclip}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPushPin}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPushPin}{\endLWR@attachfile@icon}{}{}
```

```
22
23 \xpretocmd{\atfi@acroTag}{\LWR@attachfile@icon}{}{}
24 \xapptocmd{\atfi@acroTag}{\endLWR@attachfile@icon}{}{}
```

Disable PDF file embedding:

```
25 \DeclareRobustCommand{\atfi@embedfile}[1]{}
```

The displayed output for an `\attachfile` reference:

```
26 \newcommand*{\LWR@attachfile@appearance}{}
27
28 \DeclareRobustCommand{\atfi@set@appearance}[1]{%
29     \def\LWR@attachfile@appearance{#1}%
30 }
```

A file annotation becomes a reference:

```
31 \DeclareRobustCommand{\atfi@insert@file@annot}[1]{%
32     \LWR@href{#1}{\LWR@attachfile@appearance}%
33 }
```

File 33 **lwarp-attachfile2.sty**

§ 142 Package

attachfile2

(*Emulates or patches code by* Heiko Oberdiek.)

attachfile2 (*Pkg*) attachfile2 is patched for use by lwarp.

⚠ Metadata is ignored for now.

for HTML output:

```
1 \LWR@ProvidesPackagePass{attachfile2}[2016/05/16]
```

Adds memory of the selected color:

```
2 \def\LWR@attachfiletwo@color{}%
3
4 \define@key{AtFi}{color}{%
5     \def\LWR@attachfiletwo@color{#1}%   lwarp
6   \HyColor@AttachfileColor{#1}%
7           \atfi@color@tex\atfi@color@inline\atfi@color@annot
8           {attachfile2}{color}%
9 }
```

Encloses each icon:

```
10 \newenvironment*{LWR@attachfile@icon}
11 {
12     \begin{lateximage}*%
13         [-attachfile-]%
14         [%
15             \detokenize\expandafter{\atfi@icon@icon}-%
16             \detokenize\expandafter{\LWR@attachfiletwo@color}%
17         ]%
18 }
```

```
{
    \end{lateximage}
}
```

Each icon is enclosed inside a LWR@attachfile@icon environment:

```
\xpretocmd{\atfi@acroGraph}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroGraph}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPaperclip}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPaperclip}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPushPin}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPushPin}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroTag}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroTag}{\endLWR@attachfile@icon}{}{}
```

Disable PDF file embedding:

```
\DeclareRobustCommand{\atfi@embedfile}[1]{}
```

The displayed output for an \attachfile reference:

```
\newcommand*{\LWR@attachfile@appearance}{}

\def\atfi@set@appearance@icon{%
    \atfi@set@appearance{\csname atfi@acro\atfi@icon@icon\endcsname}%
}

\DeclareRobustCommand{\atfi@set@appearance}[1]{%
    \def\LWR@attachfile@appearance{#1}%
}
```

A file annotation becomes a reference:

```
\DeclareRobustCommand{\atfi@insert@file@annot}[1]{%
    \LWR@href{#1}{\LWR@attachfile@appearance}%
}
```

Modified for text color:

```
\DeclareRobustCommand{\notextattachfile}[2][]{%
  \begingroup
    \atfi@setup{#1}%
    \ifatfi@print
      \leavevmode
      \begingroup
        \HyColor@UseColor\atfi@color@tex
        \LWR@textcurrentcolor{#2}%      lwarp
% \strut
      \endgroup
%     \else
%       \sbox\ltx@zero{#2\strut}%
%       \makebox[\wd0]{}%
    \fi
  \endgroup
}
```

Modified to draw the icon:

```
\DeclareRobustCommand{\noattachfile}[1][]{%
  \begingroup
    \atfi@setup{#1}%
    \atfi@set@appearance@icon
    \ifatfi@print
        \LWR@attachfile@appearance%     lwarp
%       \expandafter
%       \atfi@refxform\csname atfi@appobj@\atfi@icon@icon\endcsname
%     \else
%       \makebox[\atfi@appearancewidth]{}%
    \fi
  \endgroup
}
```

File 34 **lwarp-authblk.sty**

§ 143 Package

authblk

(*Emulates or patches code by* Patrick W. Daly.)

authblk (*Pkg*) authblk is patched for HTML.

package support
⚠ load order

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 69.8.

(*Emulates or patches code by* Patrick W. Daly.)

for HTML output:

Require that authblk be loaded before titling:

```
\IfPackageLoadedTF{titling}{
    \PackageError{lwarp-authblk}
        {Package authblk must be loaded before titling}
        {%
            Titling appends authblk's author macro,
            so authblk must be loaded first.%
        }
}
{\relax}
```

Load authblk:

```
\LWR@ProvidesPackagePass{authblk}[2001/02/27]
```

Patch to add a class for the affiliation:

```
\LetLtxMacro\LWRAB@affil\affil

\renewcommand{\affil}[2][]{%
\LWRAB@affil[#1]{\protect\InlineClass{affiliation}{#2}}
}
```

Create an HTML break for an \authorcr:

```
\renewcommand*{\authorcr}{\protect\LWR@newlinebr}
```

File 35 **lwarp-autobreak.sty**

§ 144 Package autobreak

(*Emulates or patches code by* Takahiro Ueda.)

autobreak (*Pkg*) autobreak is used as-is for SVG math, and nullified for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{autobreak}[2017/02/23]
```

For MathJax. The modified `align` environment is used for SVG math, but is reverted to its original for MathJax. (Extraneous commas were appearing in the result.)

```
\begin{warpMathJax}
\renewenvironment{autobreak}{\newcommand{\MoveEqLeft}[1]{}}{}
\let\start@align\@autobreak@oldstart@align
\let\endalign\@autobreak@oldendalign
\CustomizeMathJax{\newenvironment{autobreak}{}{}}
\CustomizeMathJax{\newcommand{\MoveEqLeft}[1][]{}}
\CustomizeMathJax{\newcommand{\everybeforeautobreak}[1]{}}
\CustomizeMathJax{\newcommand{\everyafterautobreak}[1]{}}
\end{warpMathJax}
```

File 36 **lwarp-autonum.sty**

§ 145 Package autonum

autonum (*Pkg*) autonum is ignored.

⚠ numbering, + All equations are numbered in HTML output. MathJax does not support the "+" environments.

for HTML output:

```
\LWR@ProvidesPackageDrop{autonum}[2015/01/18]

\RequirePackage{amsmath}


\newenvironment{equation+}{\equation}{\endequation}


\newenvironment{gather+}{\gather}{\endgather}

\BeforeBeginEnvironment{gather+}{\LWR@amsmathenv@@before{gather+}}

\AfterEndEnvironment{gather+}{\LWR@amsmathenv@@after}


\newenvironment{multline+}{\multline}{\endmultline}

\BeforeBeginEnvironment{multline+}{\LWR@amsmathenv@@before{multline+}}

\AfterEndEnvironment{multline+}{\LWR@amsmathenv@@after}
```

```
\newenvironment{flalign+}{\flalign}{\endflalign}

\BeforeBeginEnvironment{flalign+}{\LWR@amsmathenv@@before{flalign+}}

\AfterEndEnvironment{flalign+}{\LWR@amsmathenv@@after}


\newenvironment{align+}{\align}{\endalign}

\BeforeBeginEnvironment{align+}{\LWR@amsmathenv@@before{aline+}}

\AfterEndEnvironment{align+}{\LWR@amsmathenv@@after}


\newenvironment{alignat+}{\alignat}{\endalignat}

\BeforeBeginEnvironment{alignat+}{\LWR@amsmathenv@@before{alineat+}}

\AfterEndEnvironment{alignat+}{\LWR@amsmathenv@@after}


\newenvironment{split+}{\split}{\endsplit}
```

File 37 **lwarp-awesomebox.sty**

§ 146 Package awesomebox

(*Emulates or patches code by* Étienne Deparis.)

awesomebox (*Pkg*) awesomebox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{awesomebox}[2019/07/27]


\newcommand*{\LWR@awesomebox@boxborders}{}%
\newcommand*{\LWR@awesomebox@contentsborders}{}%

\newcommand*{\LWR@awesomebox@ruleborders}{%
    border-top: 1px solid black ;
    border-bottom: 1px solid black%
}

% \awesomebox[1:vrulecolor][2:hrule][3:title]{4:vrulewidth}{5:icon}{6:iconcolor}{7:content}
\RenewDocumentCommand \awesomebox { O{abvrulecolor} O{} o m m m +m }{%
    \begin{awesomeblock}[#1][#2][#3]{#4}{#5}{#6}
    #7
    \end{awesomeblock}
}

% \begin{awesomeblock}[1:vrulecolor][2:hrule][3:title]{4:vrulewidth}{5:icon}{6:iconcolor}
% <contents>
% \end{awesomeblock}
\RenewDocumentEnvironment{awesomeblock}{ O{abvrulecolor} O{} o m m m }
{%
    \LWR@forceminwidth{#4}%
    \convertcolorspec{named}{#1}{HTML}\LWR@tempcolor%
    \renewcommand*{\LWR@awesomebox@boxborders}{}%
    \renewcommand*{\LWR@awesomebox@contentsborders}{}%
```

```
    \ifdefstrequal{\abShortLine}{#2}{%
      \renewcommand*{\LWR@awesomebox@contentsborders}{\LWR@awesomebox@ruleborders}%
    }{}%
    \ifdefstrequal{\abLongLine}{#2}{%
      \renewcommand*{\LWR@awesomebox@boxborders}{\LWR@awesomebox@ruleborders}%
    }{}%
    \begin{BlockClass}[\LWR@awesomebox@boxborders]{awesomebox}
    \begin{BlockClass}[%
        margin-left: 2\% ;
        vertical-align: top
    ]{minipage}
        \color{#6}\Huge #5
    \end{BlockClass}
    \begin{BlockClass}[%
        width:75\% ;
        vertical-align: top ;
        padding-left: 1em ;
        \LWR@awesomebox@contentsborders ;
        border-left: \LWR@printlength{\LWR@atleastonept} %
            solid \LWR@origpound\LWR@tempcolor%
    ]{minipage}
        \IfValueTF{#3}{#3\newline}{}
}
{%
    \end{BlockClass}
    \end{BlockClass}
}
```

File 38 **lwarp-axessibility.sty**

§ 147 Package axessibility

axessibility (*Pkg*) axessibility is ignored.

for HTML output:

```
\PackageInfo{lwarp}{Using the lwarp version of package ‘axessibility’.}%
\ProvidesPackage{lwarp-axessibility}% no date is declared by the original

\newif\iftagpdfopt

\DeclareOption{accsupp}{
  \tagpdfoptfalse
}

\DeclareOption{tagpdf}{
  \tagpdfopttrue
}

\ProcessOptions\relax

\iftagpdfopt
    \RequirePackage{tagpdf}
\else
    \RequirePackage{accsupp}
\fi


\long\def\wrap#1{}
\long\def\wrapml#1{}
```

```
\long\def\wrapmlstar#1{}
\long\def\wrapmlalt#1{}
```

For MathJax. These usually will not be needed.

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\wrap}[1]{}}
\CustomizeMathJax{\newcommand{\wrapml}[1]{}}
\CustomizeMathJax{\newcommand{\wrapmlstar}[1]{}}
\CustomizeMathJax{\newcommand{\wrapmlalt}[1]{}}
\end{warpMathJax}
```

File 39 **lwarp-axodraw2.sty**

§ 148 Package axodraw2

(*Emulates or patches code by* John C. Collins, J.A.M. Vermaseren.)

axodraw2 (*Pkg*) axodraw2 is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{axodraw2}[2018/02/15]

\BeforeBeginEnvironment{axopicture}{%
    \begin{lateximage}[-axopicture-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{axopicture}{\end{lateximage}}
```

File 40 **lwarp-backnaur.sty**

§ 149 Package backnaur

(*Emulates or patches code by* Adrian P. Robson.)

backnaur (*Pkg*) backnaur is patched for use by lwarp, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{backnaur}[2019/06/18]

\renewenvironment{bnf}{\eqnarray}{\endeqnarray}
\renewenvironment{bnf*}{\csuse{eqnarray*}}{\csuse{endeqnarray*}}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\bnfpn}[1]{\langle \text{\textrm{#1}} \rangle}}
\CustomizeMathJax{\newcommand{\bnfor}{\; \mid \;}}
\CustomizeMathJax{\newcommand{\bnfsp}{\;}}
\IfPackageLoadedWithOptionsTF{backnaur}{perp}{
    \CustomizeMathJax{\newcommand{\bnfes}{\perp}}
}{
    \IfPackageLoadedWithOptionsTF{backnaur}{epsilon}{
        \CustomizeMathJax{\newcommand{\bnfes}{\epsilon}}
    }{
```

```
        \CustomizeMathJax{\newcommand{\bnfes}{\lambda}}
    }
}
\IfPackageLoadedWithOptionsTF{backnaur}{tsrm}{
    \CustomizeMathJax{\newcommand{\bnfts}[1]{\text{#1}}}
}{
    \CustomizeMathJax{\newcommand{\bnfts}[1]{\text{\texttt{#1}}}}
}
\CustomizeMathJax{\newcommand{\bnftd}[1]{\text{\textit{#1}}}}
\CustomizeMathJax{\newcommand{\bnfsk}{\dots}}
\IfPackageLoadedWithOptionsTF{backnaur}{altpo}{
    \CustomizeMathJax{\newcommand{\bnfpo}{::=}}
}{
    \CustomizeMathJax{\newcommand{\bnfpo}{\models}}
}
\CustomizeMathJax{\newcommand{\bnfprod}{\ifstar{\LWRbnfprodnn}{\LWRbnfprodyn}}}
\CustomizeMathJax{\newcommand{\LWRbnfprodyn}[2]{\bnfpn{#1} & \bnfpo & #2}}
\CustomizeMathJax{\newcommand{\LWRbnfprodnn}[2]{\nonumber \bnfpn{#1} & \bnfpo & #2}}
\CustomizeMathJax{\newcommand{\bnfmore}{\ifstar{\LWRbnfmorenn}{\LWRbnfmoreyn}}}
\CustomizeMathJax{\newcommand{\LWRbnfmoreyn}[1]{ &  & #1}}
\CustomizeMathJax{\newcommand{\LWRbnfmorenn}[1]{\nonumber &  & #1}}
\end{warpMathJax}
```

File 41 **lwarp-backref.sty**

§ 150 Package backref

(*Emulates or patches code by* David Carlisle and Sebastian Rahtz.)

backref (*Pkg*) backref is patched for use by lwarp.

⚠ loading Note that backref must be explicitly loaded, and is not automatically loaded by hyperref when generating HTML output.

for HTML output:

```
\LWR@ProvidesPackagePass{backref}[2016/05/21]
```

Force the hyperref option:

```
\def\backref{}

\long\def\hyper@section@backref#1#2#3{%
    \LWR@refwithsection{#3}%
}

\let\backrefxxx\hyper@section@backref
```

File 42 **lwarp-balance.sty**

§ 151 Package balance

(*Emulates or patches code by* Patrick W. Daly.)

balance (*Pkg*) balance is ignored.

for HTML output:

Discard all options for lwarp-balance:

```
\LWR@ProvidesPackageDrop{balance}[1999/02/23]

\newcommand*{\balance}{}
\newcommand*{\nobalance}{}
```

File 43 **lwarp-bbding.sty**

§ 152 Package bbding

(*Emulates or patches code by* Karel Horak, Peter Møller Neergaard.)

bbding (*Pkg*) bbding is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bbding}[1999/04/15]

\newcommand*{\LWR@bbdingsymbol}[2]{\HTMLunicode{#2}}

\newcommand{\LWR@HTML@ScissorRightBrokenBottom}{\LWR@bbdingsymbol{000}   {2701}}
\newcommand{\LWR@HTML@ScissorRight}{\LWR@bbdingsymbol{001}           {2702}}
\newcommand{\LWR@HTML@ScissorRightBrokenTop}{\LWR@bbdingsymbol{002}     {2703}}
\newcommand{\LWR@HTML@ScissorLeftBrokenBottom}{\LWR@bbdingsymbol{003}   {2701}}
\newcommand{\LWR@HTML@ScissorLeft}{\LWR@bbdingsymbol{004}            {2702}}
\newcommand{\LWR@HTML@ScissorLeftBrokenTop}{\LWR@bbdingsymbol{005}     {2703}}
\newcommand{\LWR@HTML@ScissorHollowRight}{\LWR@bbdingsymbol{006}       {2704}}
\newcommand{\LWR@HTML@ScissorHollowLeft}{\LWR@bbdingsymbol{007}       {2704}}
\newcommand{\LWR@HTML@Phone}{\LWR@bbdingsymbol{010}                   {260E}}
\newcommand{\LWR@HTML@PhoneHandset}{\LWR@bbdingsymbol{011}            {2706}}
\newcommand{\LWR@HTML@Tape}{\LWR@bbdingsymbol{012}                    {2707}}
\newcommand{\LWR@HTML@Plane}{\LWR@bbdingsymbol{013}                   {2708}}
\newcommand{\LWR@HTML@Envelope}{\LWR@bbdingsymbol{014}                {2709}}
\newcommand{\LWR@HTML@HandCuffRight}{\LWR@bbdingsymbol{015}           {261B}}
\newcommand{\LWR@HTML@HandCuffLeft}{\LWR@bbdingsymbol{016}            {261A}}
\newcommand{\LWR@HTML@HandCuffRightUp}{\LWR@bbdingsymbol{017}         {261D}}
\newcommand{\LWR@HTML@HandCuffLeftUp}{\LWR@bbdingsymbol{020}          {261F}}
\newcommand{\LWR@HTML@HandRight}{\LWR@bbdingsymbol{021}               {261E}}
\newcommand{\LWR@HTML@HandLeft}{\LWR@bbdingsymbol{022}                {261C}}
\newcommand{\LWR@HTML@HandRightUp}{\LWR@bbdingsymbol{023}             {261D}}
\newcommand{\LWR@HTML@HandLeftUp}{\LWR@bbdingsymbol{024}              {261F}}
\newcommand{\LWR@HTML@Peace}{\LWR@bbdingsymbol{025}                   {270C}}
\newcommand{\LWR@HTML@HandPencilLeft}{\LWR@bbdingsymbol{026}          {270D}}
\newcommand{\LWR@HTML@PencilRight}{\LWR@bbdingsymbol{027}             {270F}}
\newcommand{\LWR@HTML@PencilLeft}{\LWR@bbdingsymbol{030}              {270F}}
\newcommand{\LWR@HTML@PencilRightUp}{\LWR@bbdingsymbol{031}           {2710}}
\newcommand{\LWR@HTML@PencilLeftUp}{\LWR@bbdingsymbol{032}            {2710}}
\newcommand{\LWR@HTML@PencilRightDown}{\LWR@bbdingsymbol{033}         {270E}}
\newcommand{\LWR@HTML@PencilLeftDown}{\LWR@bbdingsymbol{034}          {270E}}
\newcommand{\LWR@HTML@NibRight}{\LWR@bbdingsymbol{035}                {2711}}
\newcommand{\LWR@HTML@NibLeft}{\LWR@bbdingsymbol{036}                 {2711}}
\newcommand{\LWR@HTML@NibSolidRight}{\LWR@bbdingsymbol{037}           {2712}}
\newcommand{\LWR@HTML@NibSolidLeft}{\LWR@bbdingsymbol{040}            {2712}}
\newcommand{\LWR@HTML@Checkmark}{\LWR@bbdingsymbol{041}               {2713}}
\newcommand{\LWR@HTML@CheckmarkBold}{\LWR@bbdingsymbol{042}           {2714}}
\newcommand{\LWR@HTML@XSolid}{\LWR@bbdingsymbol{043}                  {2715}}
\newcommand{\LWR@HTML@XSolidBold}{\LWR@bbdingsymbol{044}              {2716}}
\newcommand{\LWR@HTML@XSolidBrush}{\LWR@bbdingsymbol{045}             {2717}}
```

```
\newcommand{\LWR@HTML@PlusOutline}{\LWR@bbdingsymbol{046}            {2719}}
\newcommand{\LWR@HTML@Plus}{\LWR@bbdingsymbol{047}                   {271A}}
\newcommand{\LWR@HTML@PlusCenterOpen}{\LWR@bbdingsymbol{050}         {271C}}
\newcommand{\LWR@HTML@PlusThinCenterOpen}{\LWR@bbdingsymbol{051}      {271B}}
\newcommand{\LWR@HTML@Cross}{\LWR@bbdingsymbol{052}                  {271D}}
\newcommand{\LWR@HTML@CrossOpenShadow}{\LWR@bbdingsymbol{053}        {271E}}
\newcommand{\LWR@HTML@CrossOutline}{\LWR@bbdingsymbol{054}           {271F}}
\newcommand{\LWR@HTML@CrossBoldOutline}{\LWR@bbdingsymbol{055}       {271F}}
\newcommand{\LWR@HTML@CrossMaltese}{\LWR@bbdingsymbol{056}           {2720}}
\newcommand{\LWR@HTML@DavidStarSolid}{\LWR@bbdingsymbol{057}         {2721}}
\newcommand{\LWR@HTML@DavidStar}{\LWR@bbdingsymbol{060}              {2721}}
\newcommand{\LWR@HTML@FourAsterisk}{\LWR@bbdingsymbol{061}           {2722}}
\newcommand{\LWR@HTML@JackStar}{\LWR@bbdingsymbol{062}               {2723}}
\newcommand{\LWR@HTML@JackStarBold}{\LWR@bbdingsymbol{063}           {2724}}
\newcommand{\LWR@HTML@CrossClowerTips}{\LWR@bbdingsymbol{064}        {2725}}
\newcommand{\LWR@HTML@FourStar}{\LWR@bbdingsymbol{065}               {2726}}
\newcommand{\LWR@HTML@FourStarOpen}{\LWR@bbdingsymbol{066}           {2727}}
\newcommand{\LWR@HTML@FiveStarLines}{\LWR@bbdingsymbol{067}          {2729}}
\newcommand{\LWR@HTML@FiveStar}{\LWR@bbdingsymbol{070}               {2605}}
\newcommand{\LWR@HTML@FiveStarOpen}{\LWR@bbdingsymbol{071}           {2729}}
\newcommand{\LWR@HTML@FiveStarOpenCircled}{\LWR@bbdingsymbol{072}     {272A}}
\newcommand{\LWR@HTML@FiveStarCenterOpen}{\LWR@bbdingsymbol{073}      {272B}}
\newcommand{\LWR@HTML@FiveStarOpenDotted}{\LWR@bbdingsymbol{074}      {272C}}
\newcommand{\LWR@HTML@FiveStarOutline}{\LWR@bbdingsymbol{075}         {272D}}
\newcommand{\LWR@HTML@FiveStarOutlineHeavy}{\LWR@bbdingsymbol{076}    {272E}}
\newcommand{\LWR@HTML@FiveStarConvex}{\LWR@bbdingsymbol{077}         {272F}}
\newcommand{\LWR@HTML@FiveStarShadow}{\LWR@bbdingsymbol{100}         {2730}}
\newcommand{\LWR@HTML@AsteriskBold}{\LWR@bbdingsymbol{101}           {2731}}
\newcommand{\LWR@HTML@AsteriskCenterOpen}{\LWR@bbdingsymbol{102}     {2732}}
\newcommand{\LWR@HTML@AsteriskThin}{\LWR@bbdingsymbol{103}           {273B}}
\newcommand{\LWR@HTML@AsteriskThinCenterOpen}{\LWR@bbdingsymbol{104}   {273C}}
\newcommand{\LWR@HTML@EightStarTaper}{\LWR@bbdingsymbol{105}         {2733}}
\newcommand{\LWR@HTML@EightStarConvex}{\LWR@bbdingsymbol{106}        {2735}}
\newcommand{\LWR@HTML@SixStar}{\LWR@bbdingsymbol{107}                {2736}}
\newcommand{\LWR@HTML@EightStar}{\LWR@bbdingsymbol{110}              {2737}}
\newcommand{\LWR@HTML@EightStarBold}{\LWR@bbdingsymbol{111}          {2738}}
\newcommand{\LWR@HTML@TwelweStar}{\LWR@bbdingsymbol{112}             {2739}}
\newcommand{\LWR@HTML@SixteenStarLight}{\LWR@bbdingsymbol{113}       {273A}}
\newcommand{\LWR@HTML@SixFlowerPetalRemoved}{\LWR@bbdingsymbol{114}   {273B}}
\newcommand{\LWR@HTML@SixFlowerOpenCenter}{\LWR@bbdingsymbol{115}     {273C}}
\newcommand{\LWR@HTML@Asterisk}{\LWR@bbdingsymbol{116}               {273D}}
\newcommand{\LWR@HTML@SixFlowerAlternate}{\LWR@bbdingsymbol{117}      {273E}}
\newcommand{\LWR@HTML@FiveFlowerPetal}{\LWR@bbdingsymbol{120}        {273F}}
\newcommand{\LWR@HTML@SixFlowerPetalDotted}{\LWR@bbdingsymbol{121}    {2740}}
\newcommand{\LWR@HTML@FiveFlowerOpen}{\LWR@bbdingsymbol{122}         {2740}}
\newcommand{\LWR@HTML@EightFlowerPetal}{\LWR@bbdingsymbol{123}       {2741}}
\newcommand{\LWR@HTML@SunshineOpenCircled}{\LWR@bbdingsymbol{124}     {2742}}
\newcommand{\LWR@HTML@SixFlowerAltPetal}{\LWR@bbdingsymbol{125}      {2743}}
\newcommand{\LWR@HTML@FourClowerOpen}{\LWR@bbdingsymbol{126}         {273F}}
\newcommand{\LWR@HTML@FourClowerSolid}{\LWR@bbdingsymbol{127}        {273F}}
\newcommand{\LWR@HTML@AsteriskRoundedEnds}{\LWR@bbdingsymbol{130}     {2749}}
\newcommand{\LWR@HTML@EightFlowerPetalRemoved}{\LWR@bbdingsymbol{131}  {274A}}
\newcommand{\LWR@HTML@EightAsterisk}{\LWR@bbdingsymbol{132}          {274B}}
\newcommand{\LWR@HTML@SixFlowerRemovedOpenPetal}{\LWR@bbdingsymbol{133}  {2740}}
\newcommand{\LWR@HTML@SparkleBold}{\LWR@bbdingsymbol{134}            {2748}}
\newcommand{\LWR@HTML@Sparkle}{\LWR@bbdingsymbol{135}                {2747}}
\newcommand{\LWR@HTML@SnowflakeChevron}{\LWR@bbdingsymbol{136}       {2744}}
\newcommand{\LWR@HTML@SnowflakeChevronBold}{\LWR@bbdingsymbol{137}    {2746}}
\newcommand{\LWR@HTML@Snowflake}{\LWR@bbdingsymbol{140}              {2744}}
\newcommand{\LWR@HTML@CircleSolid}{\LWR@bbdingsymbol{141}            {25CF}}
```

```
\newcommand{\LWR@HTML@Ellipse}{\LWR@bbdingsymbol{142}                {274D}}
\newcommand{\LWR@HTML@EllipseSolid}{\LWR@bbdingsymbol{143}           {25CF}}
\newcommand{\LWR@HTML@CircleShadow}{\LWR@bbdingsymbol{144}           {274D}}
\newcommand{\LWR@HTML@EllipseShadow}{\LWR@bbdingsymbol{145}          {274D}}
\newcommand{\LWR@HTML@Square}{\LWR@bbdingsymbol{146}                 {25A1}}
\newcommand{\LWR@HTML@SquareSolid}{\LWR@bbdingsymbol{147}            {25A0}}
\newcommand{\LWR@HTML@SquareShadowBottomRight}{\LWR@bbdingsymbol{150}   {2751}}
\newcommand{\LWR@HTML@SquareShadowTopRight}{\LWR@bbdingsymbol{151}     {2752}}
\newcommand{\LWR@HTML@SquareShadowTopLeft}{\LWR@bbdingsymbol{152}      {2752}}
\newcommand{\LWR@HTML@SquareCastShadowBottomRight}{\LWR@bbdingsymbol{153}{2751}}
\newcommand{\LWR@HTML@SquareCastShadowTopRight}{\LWR@bbdingsymbol{154}   {2752}}
\newcommand{\LWR@HTML@SquareCastShadowTopLeft}{\LWR@bbdingsymbol{155}   {2752}}
\newcommand{\LWR@HTML@TriangleUp}{\LWR@bbdingsymbol{156}             {25B2}}
\newcommand{\LWR@HTML@TriangleDown}{\LWR@bbdingsymbol{157}           {25BC}}
\newcommand{\LWR@HTML@DiamondSolid}{\LWR@bbdingsymbol{160}           {25C6}}
\newcommand{\LWR@HTML@OrnamentDiamondSolid}{\LWR@bbdingsymbol{161}     {2756}}
\newcommand{\LWR@HTML@HalfCircleRight}{\LWR@bbdingsymbol{162}         {25D7}}
\newcommand{\LWR@HTML@HalfCircleLeft}{\LWR@bbdingsymbol{163}          {25D6}}
\newcommand{\LWR@HTML@RectangleThin}{\LWR@bbdingsymbol{164}           {2758}}
\newcommand{\LWR@HTML@Rectangle}{\LWR@bbdingsymbol{165}               {2759}}
\newcommand{\LWR@HTML@RectangleBold}{\LWR@bbdingsymbol{166}           {275A}}
\newcommand{\LWR@HTML@ArrowBoldRightStrobe}{\LWR@bbdingsymbol{167}     {27A0}}
\newcommand{\LWR@HTML@ArrowBoldUpRight}{\LWR@bbdingsymbol{170}        {27A6}}
\newcommand{\LWR@HTML@ArrowBoldDownRight}{\LWR@bbdingsymbol{171}       {27A5}}
\newcommand{\LWR@HTML@ArrowBoldRightShort}{\LWR@bbdingsymbol{172}      {27A7}}
\newcommand{\LWR@HTML@ArrowBoldRightCircled}{\LWR@bbdingsymbol{173}    {27B2}}


\LWR@formatted{ScissorRightBrokenBottom}
\LWR@formatted{ScissorRight}
\LWR@formatted{ScissorRightBrokenTop}
\LWR@formatted{ScissorLeftBrokenBottom}
\LWR@formatted{ScissorLeft}
\LWR@formatted{ScissorLeftBrokenTop}
\LWR@formatted{ScissorHollowRight}
\LWR@formatted{ScissorHollowLeft}
\LWR@formatted{Phone}
\LWR@formatted{PhoneHandset}
\LWR@formatted{Tape}
\LWR@formatted{Plane}
\LWR@formatted{Envelope}
\LWR@formatted{HandCuffRight}
\LWR@formatted{HandCuffLeft}
\LWR@formatted{HandCuffRightUp}
\LWR@formatted{HandCuffLeftUp}
\LWR@formatted{HandRight}
\LWR@formatted{HandLeft}
\LWR@formatted{HandRightUp}
\LWR@formatted{HandLeftUp}
\LWR@formatted{Peace}
\LWR@formatted{HandPencilLeft}
\LWR@formatted{PencilRight}
\LWR@formatted{PencilLeft}
\LWR@formatted{PencilRightUp}
\LWR@formatted{PencilLeftUp}
\LWR@formatted{PencilRightDown}
\LWR@formatted{PencilLeftDown}
\LWR@formatted{NibRight}
\LWR@formatted{NibLeft}
\LWR@formatted{NibSolidRight}
```

```
\LWR@formatted{NibSolidLeft}
\LWR@formatted{Checkmark}
\LWR@formatted{CheckmarkBold}
\LWR@formatted{XSolid}
\LWR@formatted{XSolidBold}
\LWR@formatted{XSolidBrush}
\LWR@formatted{PlusOutline}
\LWR@formatted{Plus}
\LWR@formatted{PlusCenterOpen}
\LWR@formatted{PlusThinCenterOpen}
\LWR@formatted{Cross}
\LWR@formatted{CrossOpenShadow}
\LWR@formatted{CrossOutline}
\LWR@formatted{CrossBoldOutline}
\LWR@formatted{CrossMaltese}
\LWR@formatted{DavidStarSolid}
\LWR@formatted{DavidStar}
\LWR@formatted{FourAsterisk}
\LWR@formatted{JackStar}
\LWR@formatted{JackStarBold}
\LWR@formatted{CrossClowerTips}
\LWR@formatted{FourStar}
\LWR@formatted{FourStarOpen}
\LWR@formatted{FiveStarLines}
\LWR@formatted{FiveStar}
\LWR@formatted{FiveStarOpen}
\LWR@formatted{FiveStarOpenCircled}
\LWR@formatted{FiveStarCenterOpen}
\LWR@formatted{FiveStarOpenDotted}
\LWR@formatted{FiveStarOutline}
\LWR@formatted{FiveStarOutlineHeavy}
\LWR@formatted{FiveStarConvex}
\LWR@formatted{FiveStarShadow}
\LWR@formatted{AsteriskBold}
\LWR@formatted{AsteriskCenterOpen}
\LWR@formatted{AsteriskThin}
\LWR@formatted{AsteriskThinCenterOpen}
\LWR@formatted{EightStarTaper}
\LWR@formatted{EightStarConvex}
\LWR@formatted{SixStar}
\LWR@formatted{EightStar}
\LWR@formatted{EightStarBold}
\LWR@formatted{TwelweStar}
\LWR@formatted{SixteenStarLight}
\LWR@formatted{SixFlowerPetalRemoved}
\LWR@formatted{SixFlowerOpenCenter}
\LWR@formatted{Asterisk}
\LWR@formatted{SixFlowerAlternate}
\LWR@formatted{FiveFlowerPetal}
\LWR@formatted{SixFlowerPetalDotted}
\LWR@formatted{FiveFlowerOpen}
\LWR@formatted{EightFlowerPetal}
\LWR@formatted{SunshineOpenCircled}
\LWR@formatted{SixFlowerAltPetal}
\LWR@formatted{FourClowerOpen}
\LWR@formatted{FourClowerSolid}
\LWR@formatted{AsteriskRoundedEnds}
\LWR@formatted{EightFlowerPetalRemoved}
\LWR@formatted{EightAsterisk}
\LWR@formatted{SixFlowerRemovedOpenPetal}
```

```
\LWR@formatted{SparkleBold}
\LWR@formatted{Sparkle}
\LWR@formatted{SnowflakeChevron}
\LWR@formatted{SnowflakeChevronBold}
\LWR@formatted{Snowflake}
\LWR@formatted{CircleSolid}
\LWR@formatted{Ellipse}
\LWR@formatted{EllipseSolid}
\LWR@formatted{CircleShadow}
\LWR@formatted{EllipseShadow}
\LWR@formatted{Square}
\LWR@formatted{SquareSolid}
\LWR@formatted{SquareShadowBottomRight}
\LWR@formatted{SquareShadowTopRight}
\LWR@formatted{SquareShadowTopLeft}
\LWR@formatted{SquareCastShadowBottomRight}
\LWR@formatted{SquareCastShadowTopRight}
\LWR@formatted{SquareCastShadowTopLeft}
\LWR@formatted{TriangleUp}
\LWR@formatted{TriangleDown}
\LWR@formatted{DiamondSolid}
\LWR@formatted{OrnamentDiamondSolid}
\LWR@formatted{HalfCircleRight}
\LWR@formatted{HalfCircleLeft}
\LWR@formatted{RectangleThin}
\LWR@formatted{Rectangle}
\LWR@formatted{RectangleBold}
\LWR@formatted{ArrowBoldRightStrobe}
\LWR@formatted{ArrowBoldUpRight}
\LWR@formatted{ArrowBoldDownRight}
\LWR@formatted{ArrowBoldRightShort}
\LWR@formatted{ArrowBoldRightCircled}
```

File 44 **lwarp-beamerarticle.sty**

§ 153 Package

beamerarticle

(*Emulates or patches code by* Till Tantau, Vedran Miletić, Louis Stuart, Joseph Wright.)

`beamerarticle` (*Pkg*) beamerarticle is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{beamerarticle}[2021/05/26]

\renewcommand<>{\textcolor}{\only#1{\beameroriginal{\textcolor}}}

\AtBeginDocument{

\renewcommand<>{\LWR@listitem}{%
    \only#1{%
        \beameroriginal{\LWR@listitem}%
    }%
}

\renewcommand<>{\LWR@itemizeitem}{%
    \only#1{%
        \beameroriginal{\LWR@itemizeitem}%
    }%
```

```
}

\renewcommand<>{\LWR@descitem}{%
    \only#1{%
        \beameroriginal{\LWR@descitem}%
    }%
}

\renewcommand<>{\abstract}{%
    \only#1{%
        \beameroriginal{\abstract}%
    }%
}

\renewcommand<>{\LWR@includegraphicsb}{%
    \only#1{%
        \beameroriginal{\LWR@includegraphicsb}%
    }%
}

\xpretocmd\frame
    {
            \LWR@forcenewpage
            \BlockClass{beamerframe}%
    }
    {}
    {\LWR@patcherror{beamerarticle}{frame}}

\xapptocmd\beamer@endframe
    {\endBlockClass}
    {}
    {\LWR@patcherror{beamerarticle}{beamer@endframe}}
```

An example in the beamer docs for `\includegraphics` shows the use of `\llap` in a frame.

```
\xpretocmd\beamer@article@startframe
    {\LWR@nulllistfills}
    {}
    {\LWR@patcherror{beamerarticle}{beamer@article@startframe}}

}% AtBeginDocument

\let\beamer@@tmpop@frametitle@default\relax
\defbeamertemplate<article>*{frametitle}{default}{%
    \paragraph*{\insertframetitle}\ \par%
    \ifdefempty{\insertframesubtitle}{}{%
        \noindent\emph{\insertframesubtitle}\par%
    }%
}


\NewDocumentCommand{\LWR@beamer@itemize}{o}{%
    \LWR@itemizestart\LWR@origitemize%
}%
\NewDocumentCommand{\LWR@beamer@description}{o o}{%
    \LWR@descriptionstart\LWR@origdescription%
}%

\xapptocmd{\LWR@patchlists}
```

```
    {%
        \LetLtxMacro\itemize\LWR@beamer@itemize%
        \LetLtxMacro\description\LWR@beamer@description%
    }
    {}
    {\LWR@patcherror{beamerarticle}{LWR@patchlists}}


\LetLtxMacro\maketitle\LWR@maketitle

\renewcommand{\subtitle}[2][]{
    \gdef\@subtitle{#2}
    \def\insertsubtitle{#2}
}
```

Add subtitle if not already present:

```
\AtBeginDocument{
\IfPackageLoadedTF{lwarp-scrextend}
    {}% komascript already has subtitle
    {% not komascript
        \xpatchcmd{\@maketitle}
            {%
                \LWR@htmltag{\LWR@tagtitleend}%
                \LWR@startpars%
            }%
            {%
                \LWR@htmltag{\LWR@tagtitleend}%
                \ifdefvoid{\@subtitle}{}{%
                    \begin{BlockClass}{subtitle}%
                    \@subtitle%
                    \end{BlockClass}%
                }%
                \LWR@startpars%
            }%
            {}
            {\LWR@patcherror{beamerarticle}{@maketitle}}
    }% not komascript
}

\RequirePackage{fancyvrb}
\DefineVerbatimEnvironment{semiverbatim}{Verbatim}{commandchars=\\\{\}}
```

File 45 **lwarp-biblatex.sty**

§ 154 Package biblatex

(*Emulates or patches code by* Philipp Lehman.)

biblatex (*Pkg*) When biblatex is used, modifications from newfloat may have to be undone.

for HTML output:

1. lwarp uses newfloat.
2. For classes with chapters which newfloat does not know about, such as CTEX-related classes, newfloat may modify \addtocontents.

3. biblatex, though, wants to patch \addtocontents, which causes an error if \addtocontents has been changed.
4. Therefore, \addtocontents is restored to its original here, since biblatex is about to be loaded.
5. This means that the newfloat's chapterlistsgaps option may no longer work.

```
\ifdef{\newfloat@addtocontents@ORI}{
    \let\addtocontents\newfloat@addtocontents@ORI
}{}
```

hyperref emulation is loaded \AtBeginDocument to avoid an options clash.

```
\AtBeginDocument{\RequirePackage{hyperref}}

\LWR@ProvidesPackagePass{biblatex}[2018/03/04]
```

The following create hyperlinks to the references. The original code to use hyperref is recreated here, because hyperref is emulated.

```
\AfterPreamble{

\let\blx@anchors\@empty
\protected\def\blx@anchor{%
    \xifinlist{\the\c@refsection @\abx@field@entrykey}{\blx@anchors}
        {}
        {\listxadd\blx@anchors{\the\c@refsection @\abx@field@entrykey}%
        \hypertarget{cite.\the\c@refsection @\abx@field@entrykey}{}}}

\protected\def\blx@imc@bibhyperref{%
    \@ifnextchar[%]
      {\blx@bibhyperref}
      {\blx@bibhyperref[\abx@field@entrykey]}}%

\long\def\blx@bibhyperref[#1]#2{%
%        \blx@sfsave
        \hyperlink{cite.\the\c@refsection @#1}{%
%             \blx@sfrest
         #2%
%        \blx@sfsave
        }%
% \blx@sfrest%
}%% \def\blx@nohyperref[#1]#2{#2}%

\protected\long\def\blx@imc@bibhyperlink#1#2{%
%        \blx@sfsave
        \hyperlink{cite.\the\c@refsection:#1}{%
%          \blx@sfrest
         #2%
%        \blx@sfsave
        }%
%          \blx@sfrest%
}%

\protected\long\def\blx@imc@bibhypertarget#1#2{%
%        \blx@sfsave%
        \hypertarget{cite.\the\c@refsection:#1}{%
```

```
%         \blx@sfrest
          #2%
%         \blx@sfsave%
         }%
%         \blx@sfrest%
}

\let\blx@imc@ifhyperref\@firstoftwo
```

Ensure that an autopage reference is current where each `\cite` is used, although this is nullified inside footnotes since they now use a LaTeX box.

```
\xpretocmd{\blx@citecmdinit}
    {\LWR@newautopagelabel{page}}%
    {}
    {\LWR@patcherror{biblatex}{blx@citecmdinit}}
```

Ensure that an autopage reference is current for each backref. If the citation is in a footnote, the backref will point to whatever preceeded the footnotes.

```
\xpatchcmd{\blx@addbackref@i}
    {\thepage}
    {\theLWR@previousautopagelabel}% ref to the most recent object
    {}
    {\LWR@patcherror{biblatex}{blx@addbackref@i A}}

\xpatchcmd{\blx@addbackref@i}
    {\c@page}
    {\c@LWR@previousautopagelabel}% ref to the most recent object
    {}
    {\LWR@patcherror{biblatex}{blx@addbackref@i B}}
```

The following patches are for back page references.

```
\DeclareListFormat{pageref}{%
  \ifnumless{\abx@pagerefstyle}{0}
    {\usebibmacro{list:plain}%
     \ifhyperref
       {%
%             \hyperlink{page.#1}{#1}%
            \LWR@refwithsection{\BaseJobname-autopage-#1}%  lwarp
       }
       {#1}}
    {\ifnumequal{\value{listcount}}{1}
       {\usebibmacro{pageref:init}}
       {}%
     \usebibmacro{pageref:comp}{#1}%
     \ifnumequal{\value{listcount}}{\value{liststop}}
       {\usebibmacro{pageref:dump}}
       {}}}

\renewbibmacro*{pageref:comp}[1]{%
  \numdef\abx@range@prev{\abx@range@prev+1}%
  \ifinteger{#1}
    {\def\abx@range@num{#1}%
     \def\abx@range@this{1}%
     \ifnumequal{\abx@range@this}{\abx@range@last}
       {}
       {\def\abx@range@prev{-1}}}
```

```
    {\ifrmnum{#1}
       {\numdef\abx@range@num{\rmntonum{#1}}%
        \def\abx@range@this{2}%
        \ifnumequal{\abx@range@this}{\abx@range@last}
          {}
          {\def\abx@range@prev{-1}}}
       {\undef\abx@range@num
        \def\abx@range@this{0}%
        \def\abx@range@prev{-1}}}%
  \ifdef\abx@range@num
    {\ifnumequal{\abx@range@num}{\abx@range@prev}
       {\def\abx@range@hold{#1}%
        \numdef\abx@range@diff{\abx@range@diff+1}}
       {\usebibmacro{pageref:dump}%
        \ifnumgreater{\abx@range@last}{-1}
          {\printdelim{multilistdelim}}
          {}%
        \ifhyperref
%           {\hyperlink{page.#1}{#1}}
          {\LWR@refwithsection{\BaseJobname-autopage-#1}}%  lwarp
          {#1}}%
     \edef\abx@range@prev{\abx@range@num}}
    {\usebibmacro{pageref:dump}%
     \ifnumgreater{\abx@range@last}{-1}
       {\printdelim{multilistdelim}}
       {}%
     \ifhyperref
%        {\hyperlink{page.#1}{#1}}
       {\LWR@refwithsection{\BaseJobname-autopage-#1}}%  lwarp
       {#1}%
     \def\abx@range@prev{-1}}%
  \edef\abx@range@last{\abx@range@this}}

\renewbibmacro*{pageref:dump}{%
  \ifnumgreater{\abx@range@diff}{0}
    {\ifcase\abx@pagerefstyle\relax % two
       \bibrangedash
       \ifhyperref
%          {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}%  lwarp
         {\abx@range@hold}%
     \or % three
       \ifnumless{\abx@range@diff}{2}
         {\printdelim{multilistdelim}}
         {\bibrangedash}%
       \ifhyperref
%          {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}%  lwarp
         {\abx@range@hold}%
     \or % two+
       \ifnumless{\abx@range@diff}{2}
         {\sqspace
          \ifhyperref
%             {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
            {\bibstring{sequens}}}
         {\bibrangedash
          \ifhyperref
%             {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
```

```
            {\abx@range@hold}}%
     \or % three+
       \ifnumless{\abx@range@diff}{2}
         {\sqspace
          \ifhyperref
%             {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
            {\bibstring{sequens}}}
         {\ifnumless{\abx@range@diff}{3}
            {\sqspace
             \ifhyperref
%                {\hyperlink{page.\abx@range@hold}{\bibstring{sequentes}}}
          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
               {\bibstring{sequentes}}}
            {\bibrangedash
             \ifhyperref
%                {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
               {\abx@range@hold}}}%
     \else % all+
       \ifnumless{\abx@range@diff}{2}
         {\sqspace
          \ifhyperref
%             {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
            {\bibstring{sequens}}}
         {\sqspace
          \ifhyperref
%             {\hyperlink{page.\abx@range@hold}{\bibstring{sequentes}}}
        {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
            {\bibstring{sequentes}}}%
     \fi
     \def\abx@range@diff{0}}
    {}}

}% \AfterPreamble
```

File 46 **lwarp-bibunits.sty**

§ 155 Package bibunits

(*Emulates or patches code by* Thorsten Hansen.)

`bibunits` (*Pkg*) bibunits is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bibunits}[2004/05/12]

\def\bu@bibdata{\BaseJobname}
```

File 47 **lwarp-bigdelim.sty**

§ 156 Package bigdelim

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

bigdelim (*Pkg*) bigdelim is used as-is for print or lateximage, and patched for HTML.

The delimiters are displayed in HTML by printing the delimiter, the text, and a thick border across the side of the \multirow which indicates the actual height of the delimiter. The delimiter character is given a class of ldelim or rdelim, and the default CSS sets this to font-size:200%

⚠ use \mrowcell \ldelim and \rdelim use \multirow, so \mrowcell must be used in the proper number of empty cells in the same column below \ldelim or \rdelim, but not in cells which are above or below the delimiter:

```
\begin{tabular}{lll}
<empty> & a & b \\
\ldelim{\{}{3}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
\mrowcell & g & h \\
<empty> & i & j \\
\end{tabular}
```

<-> a b

left { c d
e f
g h

<-> i j

For MATHJAX, limited emulation is provided which merely prints the delimter and optional text in the first row.

for HTML output: First, remove the temporary definitions of \ldelim and \rdelim, which were previously defined for tabular scanning in case bigdelim was not loaded:

```
\let\ldelim\relax
\let\rdelim\relax
```

Next, load the package's new definitions:

```
\LWR@ProvidesPackagePass{bigdelim}[2021/03/15]
```

\ldelim {⟨*1:delimiter*⟩} {⟨*2:#rows*⟩} [⟨*3: vmove*⟩] {⟨*4:width*⟩} [⟨*5:text*⟩]
\rdelim

```
\NewDocumentCommand{\LWR@HTML@ldelim}{m m o m O{}}{%
\renewcommand{\LWR@multirowborder}{right}%
\multirow{#2}{#4}{#5 \InlineClass{ldelim}{#1}}%
}

\LWR@formatted{ldelim}

\NewDocumentCommand{\LWR@HTML@rdelim}{m m o m O{}}{%
\renewcommand{\LWR@multirowborder}{left}%
\multirow{#2}{#4}{\InlineClass{rdelim}{#1} #5}%
}

\LWR@formatted{rdelim}
```

Limited emulation for MathJax. The delimiter is printed on the first row, along with any optional text.

```
\begin{warpMathJax}
% \ldelim ( {n}{width}[text]
\CustomizeMathJax{\newcommand{\LWRldelimtwo}[1][]{\text{#1}~\LWRbigdelim}}
\CustomizeMathJax{\newcommand{\LWRldelimone}[2][]{\LWRldelimtwo}}
\CustomizeMathJax{\def\ldelim#1#2{\def\LWRbigdelim{#1}\LWRldelimone}}
% \rdelim ) {n}{width}[text]
\CustomizeMathJax{\newcommand{\LWRrdelimtwo}[1][]{\LWRbigdelim~\text{#1}}}
\CustomizeMathJax{\newcommand{\LWRrdelimone}[2][]{\LWRrdelimtwo}}
\CustomizeMathJax{\def\rdelim#1#2{\def\LWRbigdelim{#1}\LWRrdelimone}}
\end{warpMathJax}
```

File 48 **lwarp-bigfoot.sty**

§ 157 Package bigfoot

`bigfoot` (*Pkg*) bigfoot is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{bigfoot}[2015/08/30]

\RequirePackage{manyfoot}
\RequirePackage{perpage}

\def\RestyleFootnote#1#2{}
\def\FootnoteSpecific#1{}
\def\DefineFootnoteStack#1{}
\def\PushFootnoteMark#1{}
\def\PopFootnoteMark#1{}
\def\hfootfraction{0.9}
\def\vtypefraction{0.7}
\def\FootnoteMinimum{1sp}
\def\FootnoteMainMinimum{0pt}
\newcount\bigfoottolerance
\bigfoottolerance=100
\providecommand\footnotecarryratio{2}
```

File 49 **lwarp-bigstrut.sty**

§ 158 Package bigstrut

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

`bigstrut` (*Pkg*) bigstrut is used as-is for print or `lateximage`, and patched for HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{bigstrut}[2018/08/03]

\LetLtxMacro\LWR@origbigstrut\bigstrut

\renewcommand\bigstrut[1][x]{}

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\bigstrut\LWR@origbigstrut%
```

```
}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\bigstrut}[1][]{}}
\end{warpMathJax}
```

File 50 **lwarp-bitpattern.sty**

§ 159 Package

bitpattern

(*Emulates or patches code by* Jean-Marc Bourguet.)

bitpattern (*Pkg*) bitpattern is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bitpattern}[2015/12/11]

\xpatchcmd{\bitpattern}
    {\begingroup}
    {\begin{lateximage}[-bitpattern-~\PackageDiagramAltText]}
    {}
    {\LWR@patcherror{bitpattern}{bitpattern}}

\xpatchcmd{\bp@Done}
    {\endgroup}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{bitpattern}{bp@Done}}
```

File 51 **lwarp-blowup.sty**

§ 160 Package

blowup

blowup (*Pkg*) blowup is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{blowup}[2018/01/02]

\newcommand*\blowUp[1]{}
```

File 52 **lwarp-bm.sty**

§ 161 Package

bm

(*Emulates or patches code by* David Carlisle, Frank Mittelbach.)

bm (*Pkg*) bm is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bm}[2019/07/24]
```

\DeclareBoldMathCommand must only be used in the preamble, since it adds to the MathJax setup code.

```
\begin{warpMathJax}
\LetLtxMacro\LWR@orig@DeclareBoldMathCommand\DeclareBoldMathCommand

\renewcommand\DeclareBoldMathCommand[3][bold]{%
    \LWR@orig@DeclareBoldMathCommand[#1]{#2}{#3}%
    \CustomizeMathJax{\newcommand{#2}{\boldsymbol{#3}}}%
}

\@onlypreamble\DeclareBoldMathCommand

\CustomizeMathJax{\newcommand{\bm}[1]{\boldsymbol{#1}}}
\end{warpMathJax}
```

File 53 **lwarp-booklet.sty**

§ 162 Package booklet

(*Emulates or patches code by* Peter Wilson.)

booklet (*Pkg*) booklet is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{booklet}[2009/09/02]

\newdimen\pageseplength
\newdimen\pagesepwidth
\newdimen\pagesepoffset
\newif\ifsidebyside     \sidebysidetrue
\newif\ifuselandscape   \uselandscapefalse
\newif\ifprintoption    \printoptionfalse
\newcommand*{\pagespersignature}[1]{}
\def\magstepminus#1{}
\newcommand*{\target}[3]{}
\newcommand*{\source}[3]{}
\newcommand*{\setpdftargetpages}{}
\newcommand*{\setdvipstargetpages}{}
\newcommand*{\targettopbottom}{}
\newcommand*{\twoupemptypage}{}
\newcommand*{\twoupclearpage}{}
\newcommand*{\checkforlandscape}{}
```

File 54 **lwarp-bookmark.sty**

§ 163 Package bookmark

(*Emulates or patches code by* Heiko Oberdiek.)

bookmark (*Pkg*) bookmark is ignored.

for HTML output: Discard all options for lwarp-bookmark:

```
\LWR@ProvidesPackageDrop{bookmark}[2016/05/17]
```

```
\newcommand*{\bookmarksetup}[1]{}
\newcommand*{\bookmarksetupnext}[1]{}
\newcommand*{\bookmark}[2][]{}
\newcommand*{\bookmarkdefinestyle}[2]{}
\newcommand*{\bookmarkget}[1]{}
\newcommand{\BookmarkAtEnd}[1]{}
```

File 55 **lwarp-booktabs.sty**

§ 164 Package

booktabs

(*Emulates or patches code by* Simon Fear.)

booktabs (*Pkg*) booktabs is emulated during HTML output, and used as-is during print output and inside an HTML `latexmage`.

⚠ `\cmidrule` For MathJax, emulation is provided in math mode, but `\cmidrule` trim must not be used.

for HTML output: If booktabs has already been loaded before lwarp, such as by memoir, use it as-is. If not, the lwarp core will have placed some dummy macros which should be removed before loading the actual booktabs definitions.

```
\IfPackageLoadedTF{booktabs}{}{
    \LetLtxMacro\toprule\relax
    \LetLtxMacro\midrule\relax
    \LetLtxMacro\cmidrule\cline
    \LetLtxMacro\bottomrule\relax
    \LetLtxMacro\addlinespace\relax
    \LetLtxMacro\morecmidrules\relax
    \LetLtxMacro\specialrule\relax
}
```

Next, load the booktabs package:

```
\LWR@ProvidesPackagePass{booktabs}[2019/10/08]
```

Adjust to work even if xltabular is loaded:

```
% \def\LWR@HTML@@BLTrule{\@BTnormal}
%
% \LWR@formatted{@BLTrule}
\LetLtxMacro\@BLTrule\@BTnormal


\DeclareDocumentCommand{\LWR@HTML@toprule}{o d()}%
    {%
        \IfValueTF{#1}%
            {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {%
                \ifbool{FormatWP}%
                {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
                    {\booltrue{LWR@doingtbrule}}%
            }%
    \LWR@getmynexttoken}

\LWR@expandableformatted{toprule}
```

```
\DeclareDocumentCommand{\LWR@HTML@midrule}{o d()}%
    {%
        \IfValueTF{#1}%
            {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {%
                \ifbool{FormatWP}%
                {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
                    {\defaddtocounter{LWR@hlines}{1}}%
            }%
    \LWR@getmynexttoken}

\LWR@expandableformatted{midrule}

\DeclareDocumentCommand{\LWR@HTML@cmidrule}{O{\LWR@cmidrulewidth} d() m}{%
    \LWR@docmidrule[#1](#2){#3}%
    \LWR@getmynexttoken%
}%

\LWR@expandableformatted{cmidrule}

\DeclareDocumentCommand{\LWR@HTML@bottomrule}{o d()}{%
    \IfValueTF{#1}%
        {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
        {%
            \ifbool{FormatWP}%
                {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
                 {\booltrue{LWR@doingtbrule}}%
        }%
    \LWR@getmynexttoken%
}%

\LWR@expandableformatted{bottomrule}

\DeclareDocumentCommand{\LWR@HTML@addlinespace}{o}{}%

\LWR@expandableformatted{addlinespace}

\DeclareDocumentCommand{\LWR@HTML@morecmidrules}{}{}%

\LWR@expandableformatted{morecmidrules}

\DeclareDocumentCommand{\LWR@HTML@specialrule}{m m m d()}%
  {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}\LWR@getmynexttoken}%

\LWR@expandableformatted{specialrule}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\toprule}[1][]{\hline}}
\CustomizeMathJax{\let\midrule\toprule}
\CustomizeMathJax{\let\bottomrule\toprule}
\CustomizeMathJax{\def\LWRbooktabscmidruleparen(#1)#2{}}
\CustomizeMathJax{\newcommand{\LWRbooktabscmidrulenoparen}[1]{}}
\CustomizeMathJax{\newcommand{\cmidrule}[1][]{%
    \ifnextchar(\LWRbooktabscmidruleparen\LWRbooktabscmidrulenoparen%
}}
\CustomizeMathJax{\newcommand{\morecmidrules}{}}
\CustomizeMathJax{\newcommand{\specialrule}[3]{\hline}}
```

```
\CustomizeMathJax{\newcommand{\addlinespace}[1][]{}}
\end{warpMathJax}
```

File 56 **lwarp-bophook.sty**

§ 165 Package

bophook

bophook (*Pkg*) bophook is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bophook}[2001/03/29]

\newcommand*{\AtBeginPage}[1]{}
\newcommand*{\PageLayout}[1]{}
```

File 57 **lwarp-bounddvi.sty**

§ 166 Package

bounddvi

bounddvi (*Pkg*) bounddvi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bounddvi}[2016/12/28]
```

File 58 **lwarp-boxedminipage.sty**

§ 167 Package

boxedminipage

(*Emulates or patches code by* Scott Pakin.)

boxedminipage (*Pkg*) boxedminipage is emulated for HTML, and used as-is for `lateximages`.

for HTML output:

```
\LWR@ProvidesPackagePass{boxedminipage}[2020/04/19]

\newenvironment{LWR@HTML@boxedminipage}{%
    \LWR@stoppars%
    \begin{BlockClass}{framebox}%
    \minipage%
}
{%
    \endminipage%
    \end{BlockClass}%
    \LWR@startpars%
}
\LWR@formattedenv{boxedminipage}
```

File 59 **lwarp-boxedminipage2e.sty**

§ 168 Package

boxedminipage2e

(*Emulates or patches code by* Scott Pakin.)

boxedminipage2e (*Pkg*) boxedminipage2e has been renamed boxedminipage by the author.

for HTML output: Automatically loads boxedminipage:

```
\LWR@ProvidesPackagePass{boxedminipage2e}
```

File 60 **lwarp-braket.sty**

§ 169 Package braket

(*Emulates or patches code by* Donald Arseneau.)

braket (*Pkg*) braket works as-is for HTML with SVG math. For MathJax, the MathJax extension is used.

for HTML output:

```
\LWR@ProvidesPackagePass{braket}% No date is provided by the file.

\begin{warpMathJax}
    \CustomizeMathJax{\require{braket}}
\end{warpMathJax}
```

File 61 **lwarp-breakurl.sty**

§ 170 Package breakurl

(*Emulates or patches code by* Vilar Camara Neto.)

breakurl (*Pkg*) breakurl is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{breakurl}[2013/04/10]

\LetLtxMacro\burl\LWR@url

\NewDocumentCommand{\LWR@burlaltb}{O{} +m m}{%
    \LWR@ensuredoingapar%
    \LWR@subhyperref{#2}%
    \LWR@subhyperreftext{#3}%
    \endgroup% restore catcodes
}

\newrobustcmd*{\burlalt}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@burlaltb%
}

\LetLtxMacro\urlalt\burlalt
```

File 62 **lwarp-breqn.sty**

§ 171 Package breqn

(*Emulates or patches code by* Michael J. Downes, Morten Høgholm.)

breqn (*Pkg*) breqn is patched for use by lwarp.

⚠ darray darray is not supported, and in fact does not work in the print version either.

While using MathJax, breqn objects are converted to SVG images.

for HTML output:

```
\LWR@ProvidesPackagePass{breqn}[2017/01/27]

\setkeys{breqn}{spread={5pt}}

\def\eqnumside{R}
% \def\eqnumplace{T}

\BeforeBeginEnvironment{dmath}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dmath- \MathImageAltText]
}

\AfterEndEnvironment{dmath}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dmath*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dmath*- \MathImageAltText]
}

\AfterEndEnvironment{dmath*}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dseries}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dseries- \MathImageAltText]
}

\AfterEndEnvironment{dseries}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dseries*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dseries*- \MathImageAltText]
}

\AfterEndEnvironment{dseries*}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dgroup}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
```

```
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dgroup- \MathImageAltText]
}

\AfterEndEnvironment{dgroup}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dgroup*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dgroup*- \MathImageAltText]
}

\AfterEndEnvironment{dgroup*}{
    \end{lateximage}\end{BlockClass}
}
```

File 63 **lwarp-bsheaders.sty**

§ 172 Package

bsheaders

bsheaders (*Pkg*) bsheaders is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bsheaders}[1997/10/06]
```

File 64 **lwarp-bussproofs.sty**

§ 173 Package

bussproofs

(*Emulates or patches code by* Samuel R. Buss.)

bussproofs (*Pkg*) bussproofs is used as-is for HTML, and emulated by MathJax's extension.

⚠ \DisplayProof If not using MathJax, inline proofs with \DisplayMath must be placed inside a math expression.

If using MathJax, only the prooftree environment may be used, not \DisplayProof.

for HTML output:

```
\LWR@ProvidesPackagePass{bussproofs}% no date in file

\ifbool{mathjax}{
    \CustomizeMathJax{\require{bussproofs}}

    \NewEnviron{LWR@HTML@prooftree}%
        {\LWR@doequation{\BODY}{prooftree}}%
        [\LWR@doendequation{prooftree}]
    \LWR@formattedenv{prooftree}
}{% SVG HTML
    \BeforeBeginEnvironment{prooftree}{%
        \begin{lateximage}[-bussproofs-~\PackageDiagramAltText]%
    }
    \AfterEndEnvironment{prooftree}{\end{lateximage}}
}
```

File 65 **lwarp-bxpapersize.sty**

§ 174 Package bxpapersize

bxpapersize (*Pkg*) bxpapersize is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bxpapersize}[2017/10/08]

\providecommand*\papersizesetup{\bxpapersizesetup}
\newcommand*\bxpapersizesetup[1]{}
```

File 66 **lwarp-bytefield.sty**

§ 175 Package bytefield

(*Emulates or patches code by* Scott Pakin.)

bytefield (*Pkg*) bytefield is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bytefield}[2017/09/15]

\BeforeBeginEnvironment{bytefield}{%
    \begin{lateximage}[-bytefield-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{bytefield}{\end{lateximage}}
```

File 67 **lwarp-cancel.sty**

§ 176 Package cancel

cancel (*Pkg*) cancel is used as-is for SVG math, and emulated for HTML text output.

for HTML output:

```
\LWR@origRequirePackage{lwarp-xcolor}% for \convertcolorspec
\LWR@ProvidesPackagePass{cancel}[2013/04/12]
```

\cancelto is math-only, so is used as-is.

\LWR@cancelcolor {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
\newcommand{\LWR@cancelcolor}[5]{%
\ifcsempty{#2}%
{\InlineClass(#5){#3}{#1}}%
{\LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}}%
}
```

\cancel {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@cancel}[1]{%
\begingroup%
\CancelColor%
\LWR@findcurrenttextcolor%
\color{black}%
\LWR@cancelcolor{#1}{LWR@tempcolor}{sout}{text-decoration-color}%
    {text-decoration:line-through}%
\endgroup%
}
\LWR@formatted{cancel}%

\LetLtxMacro\bcancel\cancel
\LetLtxMacro\xcancel\cancel
```

For MathJax:

```
\begin{warpMathJax}
\PackageNoteNoLine{lwarp, cancel}{The MathJax v3 extension will be used}
\CustomizeMathJax{\require{cancel}}
\end{warpMathJax}
```

File 68 **lwarp-canoniclayout.sty**

§ 177 Package

canoniclayout

canoniclayout (*Pkg*) canoniclayout is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{canoniclayout}[2011/11/05]

\newcommand*{\currentfontletters}{}
\newcommand*{\charactersperpage}{}
```

File 69 **lwarp-caption.sty**

§ 178 Package

caption

(*Emulates or patches code by* Axel Sommerfeldt.)

caption (*Pkg*) caption is patched for use by lwarp.

for HTML output:

```
\typeout{---}
\typeout{Packages lwarp and caption:}
\typeout{If a ‘‘Missing \protect\begin\protect{document\protect}’’ error occurs here,}
\typeout{try using: \space \protect\usepackage\protect{caption\protect}\space%
    \protect\captionsetup{options}}
\typeout{instead of: \protect\usepackage[options]\protect{caption\protect}.}
\typeout{---}

\LWR@ProvidesPackagePass{caption}[2020/10/26]

\long\def\caption@iibox@#1#2#3#4{%
```

```
%   \setbox\@tempboxa\hbox{#4}%
  \caption@iiibox{#1}{#2}{#3}%
%         [\wd\@tempboxa]%
        []%                             lwarp
        [\captionbox@hj@default]%
%         {\unhbox\@tempboxa}%
        {{#4}}%                         lwarp
}

\long\def\caption@iiiibox#1#2#3#4#5[#6][#7]#8{%
  \begingroup
  #1*% set \caption@position
  \caption@iftop{%
    \endgroup

    \minipagefullwidth%                 lwarp
    \parbox[t]{\linewidth}{%
      #1\relax
      \caption@setposition t%
%       #2%
            {\caption#4{#5}}%
%       \captionbox@hrule
%       \csname caption@hj@#7\endcsname
      #8%
    }%
  }{%
    \endgroup

%     \parbox[b]{#6}{%
    \minipagefullwidth%                 lwarp
    \parbox[b]{\linewidth}{%            lwarp
      #1\relax
      \caption@setposition b%
%       \csname caption@hj@#7\endcsname
      #8%
%       \captionbox@hrule
%       #3
            {\caption#4{#5}}%
    }%
  }%
}
```

\caption@makecaption

```
\long\def\caption@makecaption#1#2{%
%   \caption@make@above
  \caption@@make{#1}{#2}%
%   \caption@make@below
}

\AtBeginDocument{
    \let\@makecaption\caption@makecaption
}
```

Appended to look ahead to the next token for \centering, etc:

```
\AtBeginDocument{
\xapptocmd{\@xfloat}
    {\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment}
```

```
    {}
    {\LWR@patcherror{caption}{@xfloat}}

\xapptocmd{\@xdblfloat}
    {\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment}
    {}
    {\LWR@patcherror{caption}{@xdblfloat}}
}

\long\def\caption@@@text#1#2#3[#4]#5{%
    \begin{BlockClass}{figurecaption}%      lwarp
  \begingroup
    #3{\csname c@#1\endcsname #4\relax}%
    #2{\caption@fnum{#1}}{#5}%
  \endgroup%
    \end{BlockClass}%                       lwarp
}
```

Updates for late patches for scrextend:

```
\caption@AtBeginDocument{
\IfPackageLoadedTF{lwarp-scrextend}{
    \LetLtxMacro\captionbelow\caption
    \LetLtxMacro\captionabove\caption
    \LetLtxMacro\captionofbelow\captionof
    \LetLtxMacro\captionofabove\captionof
}{}
}
```

File 70 **lwarp-caption3.sty**

§ 179 Package

caption3

(*Emulates or patches code by* Axel Sommerfeldt.)

caption3 (*Pkg*) caption3 is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{caption3}[2020/10/21]
```

\caption@@@make {⟨*caption label*⟩} {⟨*caption text*⟩}

```
\IfPackageAtLeastTF{caption3}{2020/08/23}{
\renewcommand\caption@@@make[2]{%
\LWR@traceinfo{caption@@@make}%

    \LWR@stoppars%                              lwarp

%   \sbox\@tempboxa{#1}%
%   \ifdim\wd\@tempboxa=\z@
%     \caption@set{labelseparator}{none}%
%   \fi
  \caption@ifempty{#2}{%
    \caption@set{labelseparator}{none}%
    \caption@set{textformat}{simple}%
  }%
  \caption@labelseparator % defines \caption@iflabelfont,
```

```
%       \caption@labelsep and \caption@labelsep@name
%       (the latter is needed by \caption@fmt)
%
%   \@setpar{\@@par\caption@@par}\caption@@par
  \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
%   \caption@fmt
    {\ifcaption@star\else
       \begingroup
         \captionlabelfont
         \LWR@isolate{#1}%                        lwarp
       \endgroup
     \fi}%
    {\ifcaption@star\else
       \begingroup
         \caption@iflabelfont\captionlabelfont
         \relax\caption@labelsep
       \endgroup
     \fi}%
    {{\captiontextfont
        \let\\\newline%                           lwarp
%
      \caption@textstart
%       \caption@ifstrut
%         {\vrule\@height\ht\strutbox\@width\z@}%
%         {}%
%       \nobreak\hskip\z@skip % enable hyphenation
      \LWR@isolate{\caption@textformat{#2}}%      lwarp
%       \caption@ifstrut
%         {\ifhmode\@finalstrut\strutbox\fi}%
%         {}%
      \caption@textend}}%
    \LWR@startpars%                               lwarp
\LWR@traceinfo{caption@@@make done}%
}
}% later than 2020/08/23
{% earlier than 2020/08/23
\renewcommand\caption@@@make[2]{%
\LWR@traceinfo{caption@@@make}%
    \LWR@stoppars%                                lwarp
%   \sbox\@tempboxa{#1}%
%   \ifdim\wd\@tempboxa=\z@
%     \let\caption@lsep\relax
%   \fi
  \caption@ifempty{#2}{%
    \let\caption@lsep\@empty
    \let\caption@tfmt\@firstofone
  }%
%   \@setpar{\@@par\caption@@par}\caption@@par
  \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
%     \caption@fmt
```

```
    {\ifcaption@star\else
       \begingroup
        \captionlabelfont
         \LWR@isolate{#1}%                        lwarp
       \endgroup
     \fi}%
    {\ifcaption@star\else
       \begingroup
         \caption@iflf\captionlabelfont
         \relax
        \caption@lsep
       \endgroup
     \fi}%
    {{%
        \captiontextfont
        \let\\\newline%                           lwarp
%       \caption@ifstrut
%         {\vrule\@height\ht\strutbox\@width\z@}%
%         {}%
%       \nobreak\hskip\z@skip % enable hyphenation
        \LWR@isolate{\caption@tfmt{#2}}%          lwarp
%       \caption@ifstrut
%         {\ifhmode\@finalstrut\strutbox\fi}%
%         {}%
      }}%
    \LWR@startpars%                               lwarp
\LWR@traceinfo{caption@@@make done}%
}
}% earlier than 2020/08/23
```

\caption@@make@ {⟨⟩} {⟨⟩}

```
\renewcommand{\caption@@make@}[2]{%
  \caption@stepthecounter%
  \caption@beginhook%
%   \caption@box\hsize{%
%    \caption@singlelinecheck{\caption@slc{#1}{#2}\caption@singleline\caption@multiline}{\caption@mu
%      \caption@calcmargin
%      \caption@make@leftmargin
%      \caption@make@parbox{%
%        \caption@make@indention
      \caption@@@make{#1}{#2}%
%      }
%      \caption@make@rightmargin
%   }%
  \caption@endhook%
}

\DeclareCaptionBox{none}{#2}
\DeclareCaptionBox{parbox}{%
    #2%
}

\DeclareCaptionBox{colorbox}{%
    #2%
}
```

File 71 **lwarp-cases.sty**

§ 180 Package cases

(*Emulates or patches code by* Donald Arseneau.)

cases (*Pkg*) cases is patched for use by lwarp.

While using MathJax, cases objects are converted to SVG math images. The MathJax 3.2 cases package does not yet work with lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{cases}[2020/03/29]

\BeforeBeginEnvironment{numcases}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-cases- \MathImageAltText]
}

\AfterEndEnvironment{numcases}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{subnumcases}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-cases- \MathImageAltText]
}

\AfterEndEnvironment{subnumcases}{
    \end{lateximage}\end{BlockClass}
}
```

File 72 **lwarp-ccicons.sty**

§ 181 Package ccicons

(*Emulates or patches code by* Michael Ummels.)

ccicons (*Pkg*) ccicons is used as SVG images for HTML.

for HTML output: Discard all options for lwarp-ccicons:

```
\LWR@ProvidesPackagePass{ccicons}[2017/10/30]

\newcommand{\LWR@ccicons}[2]{%
    {\begin{lateximage}*[#1]\ccicons@font\char#2\end{lateximage}}
}
\renewcommand{\ccicons@logo}{\LWR@ccicons{ccLogo}{0}}
\renewcommand{\ccicons@by}{\LWR@ccicons{ccAttribution}{1}}
\renewcommand{\ccicons@sa}{\LWR@ccicons{ccShareAlike}{2}}
```

```
\renewcommand{\ccicons@nd}{\LWR@ccicons{ccNoDerivatives}{3}}
\renewcommand{\ccicons@nc}{\LWR@ccicons{ccNonCommercial}{4}}
\renewcommand{\ccicons@nceu}{\LWR@ccicons{ccNonCommercialEU}{5}}
\renewcommand{\ccicons@ncjp}{\LWR@ccicons{ccNonCommercialJP}{6}}
\renewcommand{\ccicons@pd}{\LWR@ccicons{ccPublicDomain}{7}}
\renewcommand{\ccicons@zero}{\LWR@ccicons{ccZero}{8}}
\renewcommand{\ccicons@sampling}{\LWR@ccicons{ccSampling}{9}}
\renewcommand{\ccicons@share}{\LWR@ccicons{ccShare}{10}}
\renewcommand{\ccicons@remix}{\LWR@ccicons{ccRemix}{11}}
\renewcommand{\ccicons@copy}{\LWR@ccicons{ccCopy}{12}}
\renewcommand{\ccicons@pdalt}{\LWR@ccicons{ccPublicDomainAlt}{13}}
```

File 73 **lwarp-centerlastline.sty**

§ 182 Package centerlastline

centerlastline (*Pkg*) centerlastline is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{centerlastline}[2020/10/12]

\providecommand{\centerlastline}{}
\def\endcenterlastline{\par}
```

File 74 **lwarp-centernot.sty**

§ 183 Package centernot

(*Emulates or patches code by* Heiko Oberdiek.)

centernot (*Pkg*) centernot is used as-is for SVG math, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{centernot}[2016/05/16]

\begin{warpMathJax}
\CustomizeMathJax{\require{centernot}}
\end{warpMathJax}
```

File 75 **lwarp-changebar.sty**

§ 184 Package changebar

changebar (*Pkg*) changebar is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{changebar}[2018/03/09]

\newcommand*{\cbstart}{}
\newcommand*{\cbend}{}
\newenvironment*{\changebar}{}{}
\newcommand*{\cbdelete}{}
\newcommand*{\nochnagebars}{}
\newcommand*{\cbcolor}[1]{}
```

```
\newlength{\changebarwidth}
\newlength{\deletebarwidth}
\newlength{\changebarsep}
\newcounter{changebargrey}
```

File 76 **lwarp-changelayout.sty**

§ 185 Package changelayout

(*Emulates or patches code by* Ahmed Musa.)

changelayout (*Pkg*) changelayout is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{changelayout}[2009/10/07]

\renewrobustcmd\cpl@backtodefaults{}

\renewrobustcmd\cpl@checkifoddpage{%
  \cpl@oddpagefalse%
}

\renewrobustcmd\changepagelayout[1]{%
  \setkeys[KV]{changelay}{#1}%
}

\renewrobustcmd{\changetextlayout}[1]{\changepagelayout{#1}}

\renewrobustcmd\adjustpagelayout[1]{%
  \setkeys[KV@X]{changelay}{#1}%
}

\renewrobustcmd{\adjusttextlayout}[1]{\adjustpagelayout{#1}}

\renewrobustcmd\adjusttextwidth[1]{%
  \setkeys[KV]{changelay}{#1}%
   \begin{BlockClass}[color:\LWR@colorstyle{named}{\cpl@textcolor}]{changelayout}
        \color{\cpl@textcolor}%
        \cpl@content
    \end{BlockClass}
}
```

File 77 **lwarp-changepage.sty**

§ 186 Package changepage

(*Emulates or patches code by* Peter Wilson.)

changepage (*Pkg*) changepage is ignored.

for HTML output: Discard all options for lwarp-changepage:

```
\LWR@ProvidesPackageDrop{changepage}[2009/10/20]

\newif\ifoddpage
```

```
\DeclareRobustCommand{\checkoddpage}{\oddpagetrue}
\DeclareRobustCommand{\changetext}[5]{}
\DeclareRobustCommand{\changepage}[9]{}

\@ifundefined{adjustwidth}{
\newenvironment{adjustwidth}[2]{}{}
\newenvironment{adjustwidth*}[2]{}{}
}{
\renewenvironment{adjustwidth}[2]{}{}
\renewenvironment{adjustwidth*}[2]{}{}
}


\DeclareDocumentCommand{\strictpagecheck}{}{}
\DeclareDocumentCommand{\easypagecheck}{}{}
```

File 78 **lwarp-changes.sty**

§ 187 Package

changes

(*Emulates or patches code by* Ekkart Kleinod.)

changes (*Pkg*) changes is patched for use by lwarp.

⚠ \comment Use commandnameprefix=ifneeded to avoid a conflict with the \comment command when using lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{changes}[2021/07/15]
```

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\renewcommand{\ChangesListline}[4]{%
    \IfIsInList{#1}{\Changes@loc@show}{%
        \LWR@startpars%
        #2: #3 \qquad
        \nameref{\BaseJobname-autopage-#4}%
        \LWR@stoppars%
    }{}%
}

\renewcommand{\listofchanges}[1][\@empty]{%
\setkeys{Changes@loc}{#1}%
\ifbool{Changes@optiondraft}%
{%
\IfIsInList{\Changes@loc@style}{list|summary|compactsummary}%
{}%
{%
\PackageWarning{changes}{Wrong style for list of changes: '\Changes@loc@style', using 'list' instead.}%
\def\Changes@loc@style{}%
}%
\IfIsEmpty{\Changes@loc@style}%
{\def\Changes@loc@style{list}}%
{}%
\IfStrEq{\Changes@loc@show}{all}%
{\def\Changes@loc@show{added|deleted|replaced|highlight|comment}}%
{}%
\IfIsInList{\Changes@loc@show}{added|deleted|replaced|highlight|comment}%
```

```
{}%
{%
\PackageWarning{changes}{Wrong show-value for list of changes: '\Changes@loc@show', using 'all' instead
\def\Changes@loc@show{}%
}%
\IfIsEmpty{\Changes@loc@show}%
{\def\Changes@loc@show{added|deleted|replaced|highlight|comment}}%
{}%
\IfIsEmpty{\Changes@loc@title}%
{%
\IfStrEq{\Changes@loc@style}{list}%
{\def\Changes@heading{\listofchangesname}}{}%
\IfStrEq{\Changes@loc@style}{summary}%
{\def\Changes@heading{\summaryofchangesname}}{}%
\IfStrEq{\Changes@loc@style}{compactsummary}%
{\def\Changes@heading{\compactsummaryofchangesname}}{}%
}%
{\def\Changes@heading{\Changes@loc@title}}%
\section*{\Changes@heading}
\IfIsInList{\Changes@loc@style}{list}%
{%
\IfFileExists{\jobname.\Changes@locextension}%
{%
\newread\Changes@InFile%
\openin\Changes@InFile=\jobname.\Changes@locextension%
\loop\unless\ifeof\Changes@InFile%
\read\Changes@InFile to \Changes@Line%
\ifeof\Changes@InFile\else%
\Changes@Line%
\fi
\repeat
\closein\Changes@InFile%
}{%
\emph{\changesnoloc}%
\PackageWarning{changes}{LaTeX rerun needed for list of changes}%
}%
}{}%
\IfIsInList{\Changes@loc@style}{summary|compactsummary}%
{%
\IfFileExists{\jobname.\Changes@socextension}%
{%
\newread\Changes@InFile%
\openin\Changes@InFile = \jobname.\Changes@socextension%
\loop\unless\ifeof\Changes@InFile%
\read\Changes@InFile to \Changes@Line%
\ifeof\Changes@InFile\else%
\expandafter\changes@chopline\Changes@Line\\%
\textbf{%
\IfIsColored%
{\color{\Changes@Incolor}}%
{}%
\IfIsAnonymous{\Changes@Inid}%
{%
    \LWR@textcurrentcolor{%     lwarp
        \changesauthorname: \changesanonymousname%
    }%  lwarp
}%
{%
    \LWR@textcurrentcolor{%     lwarp
\changesauthorname: \Changes@Inid%
```

```
    }%  lwarp
\IfIsEmpty{\Changes@Inname}%
{}%
{ %
    \LWR@textcurrentcolor{%     lwarp
(\Changes@Inname)%
    }%  lwarp
}%
}%
}\\%
\numdef{\Changes@InSum}{0}%
\renewcommand*{\do}[1]{%
\numdef{\Changes@InSum}{\Changes@InSum + \csuse{Changes@In########1}}%
}%
\expandafter\dopsvlist\expandafter{\Changes@loc@show}%
\ifnumcomp{\Changes@InSum}{=}{0}%
{%
% \parbox{\Changes@summary@width}{% lwarp
    \changesnochanges%
% }%    lwarp
% \\[1ex]%    lwarp
    \par%   \lwarp
}%
{%
\numdef{\Changes@InCount}{0}%
\renewcommand*{\do}[1]{%
\numdef{\Changes@InCount}{\Changes@InCount + \csuse{Changes@In########1}}%
\ifboolexpr{%
not test {\IfStrEq{\Changes@loc@style}{compactsummary}} or%
test {\ifnumgreater{\csuse{Changes@In########1}}{0}}%
}%
{%
% \parbox{\Changes@summary@width}{%  lwarp
\csuse{changes########1name}~%
% \let\cleaders\leaders\dotfill~%   lwarp
\dotfill~%   \lwarp
\csuse{Changes@In########1}%
% }%    lwarp
% \ifnumless{\Changes@InCount}{\Changes@InSum}%     lwarp
{\\}%
% {\\[1ex]}%    lwarp
}%
{}%
}%
\expandafter\dopsvlist\expandafter{\Changes@loc@show}%
    \par%   lwarp
}%
\fi%
\repeat
\closein\Changes@InFile%
}{%
\emph{\changesnosoc}%
\PackageWarning{changes}{LaTeX rerun needed for summary of changes}%
}%
}{}%
}{}%
}


\renewcommand{\Changes@Markup@comment}[3]{%
```

```
\IfStrEq{\Changes@optioncommentmarkup}{todo}%
{%
\IfIsColored%
{\colorlet{Changes@todocolor}{authorcolor}}%
{\colorlet{Changes@todocolor}{black}}%
\todo[color=Changes@todocolor!10, bordercolor=Changes@todocolor, linecolor=Changes@todocolor!70, noli
}{}%
\IfStrEq{\Changes@optioncommentmarkup}{margin}%
{%
\marginpar{%
\IfIsColored%
{\leavevmode\color{authorcolor}}%
{}%
    \LWR@textcurrentcolor{%     lwarp
\textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
    }%  lwarp
}%
}{}%
\IfStrEq{\Changes@optioncommentmarkup}{footnote}%
{%
\footnote{%
    \LWR@textcurrentcolor{%     lwarp
\textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
    }%  lwarp
}%
}{}%
\IfStrEq{\Changes@optioncommentmarkup}{uwave}%
{%
{%
\IfIsColored%
{\color{authorcolor}}%
{}%
\allowbreak%
\uwave{%
\textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
}%
}%
}{}%
}

\renewrobustcmd{\Changes@output}[7]{%
\ifbool{Changes@optiondraft}%
{%
\Changes@check@author{#2}%
\Changes@set@color{#2}%
{%
\IfIsInList{#1}{added|deleted|replaced|highlight}%
{%
\IfIsEmpty{#5}%
{%
\IfIsAuthorEmptyAtPosition{#2}{left}%
{}%
{{%
\IfIsColored%
{\color{authorcolor}}%
{}%
    \LWR@textcurrentcolor{%     lwarp
\Changes@Markup@author{\Changes@output@author@position{#2}{left}}%
    }%  lwarp
}}%
```

```
}{}%
{%
\IfStrEq{#1}{highlight}%
{}{%
\IfIsColored%
{\color{authorcolor}}%
{}%
}%
    \LWR@textcurrentcolor{%     lwarp
\IfStrEq{#1}{added}{\Changes@Markup@added{#3}}{}%
\IfStrEq{#1}{deleted}{\Changes@Markup@deleted{#4}}{}%
\IfStrEq{#1}{replaced}{{\Changes@Markup@added{#3}}\allowbreak\Changes@Markup@deleted{#4}}{}%
\IfStrEq{#1}{highlight}{\Changes@Markup@highlight{#3}}{}%
    }%  lwarp
}%
\IfIsEmpty{#5}%
{%
\IfIsAuthorEmptyAtPosition{#2}{right}%
{}%
{{%
\IfIsColored%
{\color{authorcolor}}%
{}%
    \LWR@textcurrentcolor{%     lwarp
\Changes@Markup@author{\Changes@output@author@position{#2}{right}}%
    }%  lwarp
}}%
}{}%
\stepcounter{Changes@#1Count#2}%
}{}%
\IfIsEmpty{#5}%
{}%
{%
\stepcounter{Changes@commentCount#2}%
\Changes@set@commentcount{#2}%
\Changes@Markup@comment%
{#5}%
{#2}%
{\Changes@output@author{#2}}%
}%
}%
\IfIsEmpty{#2}%
{\def\Changes@locid{}}%
{\def\Changes@locid{~(#2)}}%
\addtocontents{\Changes@locextension}{\protect\ChangesListline{#1}{#6\Changes@locid}{#7}{\thepage}}%
}%
{%
\IfIsEmpty{#3}%
{\@bsphack\@esphack}%
{#3}%
}%
}
```

File 79 **lwarp-chappg.sty**

§ 188 Package chappg

(*Emulates or patches code by* Robin Fairbairns.)

chappg (*Pkg*) chappg is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{chappg}[2006/05/09]

\renewcommand{\pagenumbering}[2][]{}
\providecommand{\chappgsep}{--}
```

File 80 **lwarp-chapterbib.sty**

§ 189 Package chapterbib

(*Emulates or patches code by* Donald Arseneau.)

chapterbib (*Pkg*) chapterbib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{chapterbib}[2010/09/18]

\xdef\@savedjobname{\BaseJobname}
\let\@currentipfile\@savedjobname
```

File 81 **lwarp-chemfig.sty**

§ 190 Package chemfig

(*Emulates or patches code by* Christian Tellechea.)

chemfig (*Pkg*) chemfig is patched for use by lwarp.

If using `\polymerdelim` to add delimiters to a `\chemfig`, wrap both inside a single `lateximage`:

```
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

The images are not hashed because they depend on external settings which may be changed at any time, and are unlikely to be reused inline anyhow.

for HTML output:

```
\LWR@ProvidesPackagePass{chemfig}[2021/02/28]

\catcode`\_=11

\IfPackageAtLeastTF{chemfig}{2020/03/05}
{
    \xpretocmd\charge{\begin{lateximage}[-chemfig-~\PackageDiagramAltText]}
        {}{\LWR@patcherror{chemfig}{charge}}
    \xpretocmd\Charge{\begin{lateximage}[-chemfig-~\PackageDiagramAltText]}
        {}{\LWR@patcherror{chemfig}{Charge}}
    \xapptocmd\charge_c{\end{lateximage}}
        {}{\LWR@patcherror{chemfig}{charge_c}}
}{}

\IfPackageAtLeastTF{chemfig}{2019/04/18}%
{% 2019/04/18 or newer
```

```
    \xpretocmd{\CF_chemfiga}
        {\begin{lateximage}[-chemfig-~\PackageDiagramAltText]}
        {}{\LWR@patcherror{chemfig}{CF_chemfiga}}

    \xpatchcmd{\CF_chemfigb}
        {\let\CF_flipstate\CF_zero}
        {\end{lateximage}\let\CF_flipstate\CF_zero}
        {}{\LWR@patcherror{chemfig}{CF_chemfigb}}

    \GlobalLetLtxMacro\LWR@chemfig@origCF_lewisc\CF_lewisc
    \gdef\CF_lewisc#1,#2\_nil{%
    \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    \LWR@chemfig@origCF_lewisc#1,#2\_nil
    \end{lateximage}
    }

    \gpreto{\schemestart}{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    }
    \gappto{\CF_schemestop}{\end{lateximage}}

}% 2019/04/18 or newer
{% older than 2019/04/18

    \LetLtxMacro\LWR@chemfig@origchemfig\chemfig

    \DeclareDocumentCommand\chemfig{s O{} O{} m}{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
        \IfBooleanTF{#1}{%
            \LWR@chemfig@origchemfig*[#2][#3]{#4}%
        }{%
            \LWR@chemfig@origchemfig[#2][#3]{#4}%
        }
        \end{lateximage}%
    }

    \LetLtxMacro\LWR@chemfig@origCF@lewis@b\CF@lewis@b

    \def\CF@lewis@b#1#2{%
    \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    \LWR@chemfig@origCF@lewis@b{#1}{#2}%
    \end{lateximage}%
    }

    \preto{\schemestart}{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    }
    \appto{\CF@schemestop}{\end{lateximage}}

}% older than 2019/04/18

\catcode`\_=8%



\LetLtxMacro\LWR@chemfig@origchemleft\chemleft

\def\chemleft#1#2\chemright#3{%
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
\LWR@chemfig@origchemleft#1#2\chemright#3%
```

```
\end{lateximage}%
}

\LetLtxMacro\LWR@chemfig@origchemup\chemup

\def\chemup#1#2\chemdown#3{%
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
\LWR@chemfig@origchemup#1#2\chemdown#3%
\end{lateximage}%
}
```

File 82 **lwarp-chemformula.sty**

§ 191 Package

chemformula

(*Emulates or patches code by* Clemens Niederberger.)

chemformula (*Pkg*)

chemformula is patched for use by lwarp.

The svg images are hashed according to contents and local options. Global options are assumed to be constant document-wide.

⚠ chemformula with MathJax

chemformula works best without MathJax. If MathJax is used, `\displaymathother` must be used before `array`, and then `\displaymathnormal` may be used after. (The chemformula package adapts to `array`, but does not know about MathJax, and MathJax does not know about chemformula.)

While using MathJax, `\displaymathother` may also be used for other forms of display and inline math which contain chemformula expressions.

for HTML output:

```
\LWR@ProvidesPackagePass{chemformula}[2022/01/23]

\ExplSyntaxOn
```

`\ch`

Enclose in an inline svg image or MathJax. The `alt` tag is is the contents of the `\ch` expression. The filename is hashed, and also has additional hashing information based on the local options.

```
\RenewDocumentCommand \ch { O{}m }
 {%
```

To work inside `align` with `\displaymathother`, a simple version must be used to work with chemformula's adaptation to `align`.

```
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%  lwarp
    {
        \chemformula_ch:nn {#1} {#2}%   original
    }
```

If used as the outer level, must temporarily ensure MathJax is disabled:

```
    {
        \begingroup%
        \boolfalse{mathjax}%
```

An inline image is used, adjusted for the baseline:

```
        \LWR@subsingledollar*{% lwarp
```

```
            \textbackslash{}%
            ch%
            \{%
                \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
            \}% alt text
        }{%
           \protect\LWR@HTMLsanitizedetokenized{%
                \detokenize\expandafter{#1}%
            }% add'l hashing
        }%
        {%
            \chemformula_ch:nn {#1} {#2}%   original
        }%
        \endgroup%
    }
  }
```

\chcpd

Similar to \ch.

```
\IfPackageAtLeastTF{chemformula}{2019/10/13}{
\cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
  {
    \begingroup%
    \boolfalse{mathjax}%
    \LWR@subsingledollar*{% lwarp
        \textbackslash{}%
        chcpd%
        \{%
            \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
        \}%
    }{%
        \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
    }{% original
    \group_begin:
      \tl_if_blank:nF {#2}
        {
          \keys_set:nn {chemformula} {#1}
          \__chemformula_save_catcodes:
          \__chemformula_sanitize:Nn
            \l__chemformula_chemformula_tmpa_tl
            {#2}
          \__chemformula_input_compound_no_check:NV
            \l__chemformula_compound_tl
            \l__chemformula_chemformula_tmpa_tl
          \__chemformula_prepare_output:NV
            \l__chemformula_compound_tl
            \l__chemformula_catcodes_tl
          \chemformula_write:V \l__chemformula_compound_tl
        }
    \group_end:
    }
    \endgroup
  }
}% later than 2019/10/13
{% earlier than 2019/10/13
% \changes{v0.903}{2021/12/18}{\pkg{chemformula}: Improved alt tag sanitization.}
\cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
  {
    \begingroup%
    \boolfalse{mathjax}%
```

```
    \LWR@subsingledollar*{% lwarp
        \textbackslash{}%
        chcpd%
        \{%
            \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
        \}%
    }{%
        \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
    }{% original
    \group_begin:
      \tl_if_blank:nF {#2}
        {
          \keys_set:nn {chemformula} {#1}
          \__chemformula_save_catcodes:
          \__chemformula_sanitize:Nn
            \l__chemformula_chemformula_tmpa_tl
            {#2}
          \__chemformula_input_compound_no_check:NV
            \l__chemformula_compound_tl
            \l__chemformula_chemformula_tmpa_tl
          \__chemformula_prepare_output:N \l__chemformula_compound_tl
          \chemformula_write:V \l__chemformula_compound_tl
        }
    \group_end:
    }
    \endgroup
  }
}% earlier than 2019/10/13
```

\charrow

If standalone, appears in a regular lateximage.

```
\RenewDocumentCommand \charrow { mO{}O{} }
{
    \begin{lateximage}[-chemformula- charrow]
    \group_begin:
      \__chemformula_draw_arrow:nnn {#1} {#2} {#3}
    \group_end:
    \end{lateximage}
}
```

\chname

If standalone, appears in a regular lateximage, hashed according to contents.

```
\RenewDocumentCommand \chname { R(){}R(){} }
  {
    \begin{lateximage}*[%
        \textbackslash{}%
        chname%
        (\LWR@HTMLsanitizedetokenized{\detokenize{#1}})%
        (\LWR@HTMLsanitizedetokenized{\detokenize{#2}})%
    ]*%
        \chemformula_chwritebelow:nn {#1} {#2}
    \end{lateximage}
  }
```

\chlewis

Placed inline, hashed according to contents and options.

```
\RenewDocumentCommand \chlewis { O{}mm }
  {
    \begingroup%
```

```
    \boolfalse{mathjax}%
    \LWR@subsingledollar*{\textbackslash{}chlewis\{#2\}\{#3\}}%
    {
        \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
    }{
        \chemformula_lewis:nnn {#1} {#2} {#3}
    }
    \endgroup%
  }
```

lwarp redefines the $ character, so special handling is required to escape math expressions inside \ch.

This boolean tracks a new kind of escaped math:

```
\bool_new:N       \l__chemformula_first_last_LWRdollar_bool
```

\chemformula_input_escape_math

Adds additional escaping for the new dollar definition:

```
\cs_gset_protected:Npn \__chemformula_input_escape_math:n #1
  {
    \__chemformula_first_last_math:n {#1}
    \bool_if:NT \l__chemformula_first_last_dollar_bool
      {
        \bool_set_true:N \l__chemformula_first_last_math_bool
        \__chemformula_read_escape_dollar:w #1 \q_nil
      }
    \bool_if:NT \l__chemformula_first_last_mathbraces_bool
      {
        \bool_set_true:N \l__chemformula_first_last_math_bool
        \__chemformula_read_escape_mathbraces:w #1 \q_nil
      }
```

Added by lwarp:

```
    \bool_if:NT \l__chemformula_first_last_LWRdollar_bool%      lwarp
      {
        \bool_set_true:N \l__chemformula_first_last_math_bool%  lwarp
        \__chemformula_read_escape_LWRdollar:w #1 \q_nil%       lwarp
      }
  }
```

\chemformula_read_escape_LWRdollar

The following parses the contents inside the new dollars.

lwarp keeps the dollar as its original math shift until the document starts. While chemmacros is being patched, the dollar must temporarily be set to its new meaning during the following definition.

```
\begingroup
\catcode`\$=\active

\cs_new_protected:Npn \__chemformula_read_escape_LWRdollar:w $#1$ \q_nil
{
    \__chemformula_read_escape_math:n {#1}
}

```

```
\endgroup
```

\chemformula_bool_set_if_first_last

The following looks at the first and last tokens for delimiters to escape math inside \ch. The original definition is modified to look for the control sequences which are used by the new meaning of $.

```
\cs_new_protected:Npn \__chemformula_bool_cs_set_if_first_last:NnNN #1#2#3#4
  {
    \int_zero:N \l__chemformula_tmpa_int
    \int_zero:N \l__chemformula_tmpb_int
    \int_set:Nn \l__chemformula_tmpa_int { \tl_count:n {#2} }
    \tl_map_inline:nn {#2}
      {
        \int_incr:N \l__chemformula_tmpb_int
        \int_compare:nT { \l__chemformula_tmpb_int = 1 }
          {
```

At the start, the cs_ version compares control sequences:

```
            \ifdefstrequal{##1}{#3}% lwarp
                {
                    \bool_set_true:N #1
                }% lwarp
                {}
          }
```

At the end, compare more control sequences:

```
      \int_compare:nT { \l__chemformula_tmpb_int = \l__chemformula_tmpa_int }
          {
            \ifdefstrequal{##1}{#4}
                {}
                {
                    \bool_set_false:N #1
                }
          }
    }
  }
```

\chemformula_first_last_math

Modified to check for the new meaning of $ at first/last:

```
\cs_gset_protected:Npn \__chemformula_first_last_math:n #1
  {
    \bool_set_false:N \l__chemformula_first_last_math_bool
    \bool_set_false:N \l__chemformula_first_last_dollar_bool
    \bool_set_false:N \l__chemformula_first_last_LWRdollar_bool%    lwarp
    \bool_set_false:N \l__chemformula_first_last_mathbraces_bool
    \__chemformula_bool_set_if_first_last:Nnnn
      \l__chemformula_first_last_dollar_bool
      {#1}
      { $ } { $ }
    \bool_if:NF \l__chemformula_first_last_dollar_bool
      {
        \__chemformula_bool_set_if_first_last:Nnnn
          \l__chemformula_first_last_mathbraces_bool
          {#1}
```

```
199          { \( } { \) }
```

Added by lwarp:

```
200          \bool_if:NF \l__chemformula_first_last_mathbraces_bool%   lwarp
201            {
202                \__chemformula_bool_cs_set_if_first_last:NnNN
203                \l__chemformula_first_last_LWRdollar_bool
204                {#1}
205                { \LWR@newsingledollar } { \LWR@newsingledollar }
206            }%  lwarp
207      }
208  }

209 \ExplSyntaxOff
```

File 83 **lwarp-chemgreek.sty**

§ 192 Package

chemgreek

(*Emulates or patches code by* Clemens Niederberger.)

chemgreek (*Pkg*) chemgreek is patched for use by lwarp.

Greek symbols
⚠ package selection

To use text-mode symbols, use packages textalpha or textgreek. Using the other packages supported by chemgreek will result in math-mode greek characters, which will result in SVG images being used. These images will be hashed.

⚠ XeLaTeX, LuaLaTeX

If using XeLaTeX or LuaLaTeX, select the fontspec mapping:

```
\selectchemgreekmapping{fontspec}
```

for HTML output:

```
1 \LWR@ProvidesPackagePass{chemgreek}[2020/01/16]

2 \ExplSyntaxOn
3
4 \cs_gset_protected:Npn \chemgreek_text:n #1
5   { { \text {#1} } }
6
7 \appto\LWR@restoreorigformatting{%
8 \cs_set_protected:Npn \chemgreek_text:n #1%
9   { \ensuremath { \text {#1} } }%
10 }
11
12 \ExplSyntaxOff
```

File 84 **lwarp-chemmacros.sty**

§ 193 Package

chemmacros

(*Emulates or patches code by* Clemens Niederberger.)

chemmacros (*Pkg*) chemmacros is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{chemmacros}[2022/02/13]
```

svg file hashing assumes that the relevent options are constant for the entire document.

§ 193.1 Changes to the user's document

⚠ `\makepolymerdelims` When using `\makepolymerdelims`, enclose the entire expression inside a `polymerdelims` environment, such as (from the chemmacros manual):

```
\begin{polymerdelims}
\chemfig{-[@{op,.75}]CH_2-CH(-[6]Cl)-[@{cl,0.25}]}
\makepolymerdelims{5pt}[27pt]{op}{cl}
\end{polymerdelims}
```

⚠ redox reactions Redox reactions must be enclosed inside a `redoxreaction` environment. For print output, extra space must be included above and/or below the result, so they are declared as arguments to the environment, instead of being manually entered as per the chemmacros manual. For HTML output, the extra space is ignored and a `lateximage` is used instead.

```
\begin{redoxreaction}{7mm}{7mm}
\OX{a,Na} $\rightarrow$ \OX{b,Na}\pch\redox(a,b){oxidation}
\end{redoxreaction}
```

§ 193.2 Code

```
\ExplSyntaxOn
```

§ 193.3 Loading packages

Also accept the lwarp version:

```
\prg_set_conditional:Npnn \chemmacros_if_package_loaded:n #1 {p,T,F,TF}
  {
    \cs_if_exist:cTF {ver@#1.sty}
      { \prg_return_true: }
      {
        \cs_if_exist:cTF {ver@lwarp-#1.sty}
            { \prg_return_true: }
            { \prg_return_false: }
      }
  }
```

Nullify hyperref detection:

```
\hook_gput_code:nnn {begindocument/end} {chemmacros}
  {
      \bool_set_false:N \l__chemmacros_hyperref_bool
  }
```

§ 193.4 Loading modules

Patching chemmacros modules must be done `\AtBeginDocument`, since modules are invoked by the user in the preamble, and each patch is only done if the module is loaded.

§ 193.5 New environments

`\makepolymerdelims` and redox reactions must be enclosed in a `lateximage` during HTML output. These environments are provided here in HTML mode, and in the lwarp core in print mode, as a high-level semantic syntax which automatically embeds the contents in a `lateximage` with an appropriate `alt` tag.

Env `polymerdelims`

```
\DeclareDocumentEnvironment{polymerdelims}{}
{\begin{lateximage}[-chemmacros- polymer]}
{\end{lateximage}}
```

Env `redoxreaction` {⟨*space above*⟩} {⟨*space below*⟩}

For HTML output, the above and below space is ignored, and a `lateximage` is used instead. For the print output version, see section 90.

```
\DeclareDocumentEnvironment{redoxreaction}{m m}
{\begin{lateximage}[-chemmacros- redoxreaction]}
{\end{lateximage}}
```

§ 193.6 Acid-base

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{acid-base}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~acid-base}

\cs_gset_protected:Npn \chemmacros_p:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{
        \textbackslash{}%
        p%
        \{%
            \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
        \}
    }{
        chemmacrosp%
        \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
    }{
    \group_begin:
      \mbox
        {
          \chemmacros_p_style:n {p}
          \ensuremath {#1}
        }
    \group_end:
    }
    \endgroup
  }

\RenewDocumentCommand \pH  {} {
```

```
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pH}{chemmacros}{
        \chemmacros_p:n { \chemmacros_formula:n {H} }
    }
    \endgroup
}

\RenewDocumentCommand \pOH {} {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pOH}{chemmacros}{
        \chemmacros_p:n { \chemmacros_formula:n {OH} }
    }
    \endgroup
}

\RenewDocumentCommand \pKa {O{}}
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pKa{[}#1{]}}{chemmacros #1}{
        \chemmacros_p:n
        {
            \Ka \ifblank {#1} {}
            { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
        }
    }
    \endgroup
  }

\RenewDocumentCommand \pKb {O{}}
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pKb{[}#1{]}}{chemmacros #1}{
        \chemmacros_p:n
        {
            \Kb \ifblank {#1} {}
            { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
        }
    }
    \endgroup
  }

\LetLtxMacro\LWR@chemmacros@origKa\Ka
\renewcommand*{\Ka}{%
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}Ka}{chemmacros}{%
        \LWR@chemmacros@origKa%
    }%
    \endgroup
}

\LetLtxMacro\LWR@chemmacros@origKb\Kb
\renewcommand*{\Kb}{%
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}Kb}{chemmacros}{%
```

```
        \LWR@chemmacros@origKb%
    }%
    \endgroup
}

\LetLtxMacro\LWR@chemmacros@origKw\Kw
\renewcommand*{\Kw}{%
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}Kw}{chemmacros}{
        \LWR@chemmacros@origKw
    }
    \endgroup
}

}{}% module loaded
}% AtBeginDocument
```

§ 193.7 Charges

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{charges}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~charges}

\cs_gset_protected:Npn \fplus  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}fplus}{chemmacros}
    { \LWR@origensuredmath{\chemformula_fplus:}  }
    \endgroup
}
\cs_gset_protected:Npn \fminus {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}fminus}{chemmacros}
    { \LWR@origensuredmath{\chemformula_fminus:} }
    \endgroup
}

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.8 Nomenclature

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{nomenclature}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~nomenclature}

\cs_gset_protected:Npn \chemmacros_charge:n #1
  {
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}
    {\chemmacros_formula:n { {}^{#1} }}
    {
        \ifmmode
            {\chemmacros_formula:n { {}^{#1} }}
        \else
            { \textsuperscript{\ensuremath{#1}} }
        \fi
    }
  }
```

```
\hook_gput_code:nnn {begindocument/end} {chemmacros}
{
\protected\def\LWR@HTML@chemprime { \HTMLunicode{2032} }
\LWR@formatted{chemprime}
}

\cs_gset_protected:Npn \__chemmacros_cip:n #1
  {
    \tl_set:Nn \l__chemmacros_tmpa_tl {#1}
    \int_step_inline:nnnn {0} {1} {9}
      {
        \tl_replace_all:Nnn \l__chemmacros_tmpa_tl
          {##1}
          { { \l__chemmacros_cip_number_tl ##1} }
      }
    {
        \l__chemmacros_cip_inner_tl
        \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
            \l__chemmacros_tmpa_tl
        }}% lwarp
    }
  }

\RenewDocumentCommand \Sconf { O{S} } {
\begin{lateximage}[\textbackslash{}Sconf{[}#1{]}]*
    \chemmacros_sconf:n {#1}
\end{lateximage}
}

\RenewDocumentCommand \Rconf { O{R} } {
\begin{lateximage}[\textbackslash{}Rconf{[}#1{]}]*
    \chemmacros_rconf:n {#1}
\end{lateximage}
}

\cs_gset_protected:Npn \chemmacros_hapto:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}hapto\{#1\}}{chemmacros}{
        \chemmacros_coordination_symbol:nnnn
        { \l__chemmacros_coord_use_hyphen_bool }
        {
            { \c_true_bool }
        }
        { \chemeta }
        {#1}
    }
    \endgroup
  }

\cs_gset_protected:Npn \chemmacros_dento:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}dento\{#1\}}{chemmacros}{
        \chemmacros_coordination_symbol:nnnn
        { \l__chemmacros_coord_use_hyphen_bool }
        {
```

```
            { \c_true_bool }
        }
        { \chemkappa }
        {#1}
    }
    \endgroup
  }

\cs_gset_protected:Npn \chemmacros_bridge:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}bridge\{#1\}}{chemmacros}{
        \chemmacros_coordination_symbol:nnnn
        { \l__chemmacros_coord_use_hyphen_bool }
        { \l__chemmacros_bridge_super_bool }
        { \chemmu }
        {#1}
    }
    \endgroup
  }
}{}% Module loaded.
}% AtBeginDocument
```

§ 193.9 Particles

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{particles}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~particles}

\cs_gset_protected:Npn \chemmacros_declare_nucleophile:Nn #1#2
  {
    \cs_set_protected:cpn {__chemmacros_ \chemmacros_remove_backslash:N #1:}
      {
        \bool_if:NTF \l__chemmacros_nucleophile_elpair_bool
          {
            \chemmacros_elpair:n { #2 }
              { \skip_horizontal:N \l__chemmacros_nucleophile_dim }
            \chemmacros_formula:n { {}^{-} }
          }
          { \chemmacros_formula:n { #2^{-} } }
      }
    \DeclareDocumentCommand #1 {o}
      {%
        \begin{lateximage}%
        \group_begin:%
          \IfNoValueF {##1}%
            { \chemmacros_set_keys:nn {particles} {##1} }%
          \use:c {__chemmacros_ \chemmacros_remove_backslash:N #1:}%
        \group_end:%
        \end{lateximage}%
      }
  }

\RenewChemNucleophile \Nuc {Nu}
\RenewChemNucleophile \ba  {ba}

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.10 Phases

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{phases}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~phases}

\cs_undefine:N \chemmacros_phase:n
\cs_new_protected:Npn \chemmacros_phase:n #1
  {

    \mode_leave_vertical:

    \bool_if:NTF \l__chemmacros_phases_sub_bool
      {
        \ifnumequal{\value{LWR@lateximagedepth}}{0}
        {
            \textsubscript{ (#1) }
        }
        {
            \chemformula_subscript:n { (#1) }
        }
      }
      {
        \skip_horizontal:N \l__chemmacros_phases_space_dim
        \chemmacros_text:n { (#1) }
      }
  }

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.11 Mechanisms

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{mechanisms}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~mechanisms}

\chemmacros_define_keys:nn {textmechanisms}
  {
    type      .choice: ,
    type /    .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{N}
          }
          { }
      } ,
    type / 1  .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{N}
            1
          }
          { }
      } ,
    type / 2  .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
```

```
          {
              \textsubscript{N}
            2
          }
          { }
      } ,
    type / se .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
          }
          { }
      } ,
    type / 1e .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
            1
          }
          { }
      } ,
    type / 2e .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
            2
          }
          { }
      } ,
    type / ar .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
          }
          { Ar - }
      } ,
    type / e  .code:n    =
      { \__chemmacros_set_mechanisms:nnn { E } { } { } } ,
    type / e1 .code:n    =
      { \__chemmacros_set_mechanisms:nnn { E } { 1 } { } } ,
    type / e2 .code:n    =
      { \__chemmacros_set_mechanisms:nnn { E } { 2 } { } } ,
    type / cb .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { E }
          {
            1
              \textsubscript{cb}
          }
          { }
      } ,
    type      .default:n =
  }

\cs_gset_protected:Npn \chemmacros_mechanisms:n #1
  {
```

```
    \tl_if_blank:nTF {#1}
      { \chemmacros_set_keys:nn {textmechanisms} { type } }
      { \chemmacros_set_keys:nn {textmechanisms} { type = #1 } }
    \mbox
      {
        \tl_use:N \l__chemmacros_mechanisms_ar_tl
        \tl_use:N \l__chemmacros_mechanisms_type_tl
        \tl_use:N \l__chemmacros_mechanisms_mol_tl
      }
  }

\appto\LWR@restoreorigformatting{%
\cs_set_protected:Npn \chemmacros_mechanisms:n #1%
  {%
    \tl_if_blank:nTF {#1}%
      { \chemmacros_set_keys:nn {mechanisms} { type } }%
      { \chemmacros_set_keys:nn {mechanisms} { type = #1 } }%
    \mbox%
      {%
        \tl_use:N \l__chemmacros_mechanisms_ar_tl%
        \tl_use:N \l__chemmacros_mechanisms_type_tl%
        \tl_use:N \l__chemmacros_mechanisms_mol_tl%
      }%
  }%
}

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.12 Newman

There are so many options that it is hard to hash these images for reuse.

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{newman}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~newman}

\RenewDocumentCommand \newman {od()m}%
  {
    \IfValueTF{#2}
    {\begin{lateximage}[\textbackslash{}newman(#2)\{#3\}]*}
    {\begin{lateximage}[\textbackslash{}newman\{#3\}]*}
    \group_begin:
      \IfNoValueF  {#1} { \chemmacros_set_keys:nn {newman} {#1} }
      \IfNoValueTF {#2}
        { \chemmacros_newman:nn {  } {#3} }
        { \chemmacros_newman:nn {#2} {#3} }
    \group_end:
    \end{lateximage}
  }%

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.13 Orbital

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{orbital}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~orbital}

\RenewDocumentCommand \orbital {om}
  {
    \IfValueTF{#1}
    {
        \begin{lateximage}[%
            \textbackslash{}%
            orbital{[}%
            \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
            {]}\{#2\}%
        ]*[][margin-left: 1em ; margin-right: 1em]
    }
    {
        \begin{lateximage}[%
            \textbackslash{}orbital\{#2\}%
        ]*[][margin-left: 1em ; margin-right: 1em]
    }
    \group_begin:
      \chemmacros_set_keys:nn {orbital/type} {#2}
      \IfNoValueTF {#1}
        { \chemmacros_orbital:n {  } }
        { \chemmacros_orbital:n {#1} }
    \group_end:
    \end{lateximage}
  }

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.14 Reactions

\chemmacros_declare_reaction_env {⟨*chem*⟩} {⟨*math*⟩} {⟨*args number*⟩} {⟨*argument list ({#2}{#3}...)*⟩}

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{reactions}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~reactions}

% #1: chem
% #2: math
% #3: args number
% #4: argument list ({#2}{#3}...)
\cs_gset_protected:Npn \__chemmacros_declare_reaction_env:nnnn #1#2#3#4
  {
    \exp_args:Nnx \DeclareDocumentEnvironment {#1}
      { \int_compare:nT { #3+0 = 0 } {!} O{} \prg_replicate:nn {#3+0} {m} }
      {
        \boolfalse{mathjax}%                        lwarp
        \ifdefvoid{\LWR@ThisAltText}{%              lwarp
            \ThisAltText{-chemmacros-~reaction}%    lwarp
        }{}%                                        lwarp
        \chemmacros_add_reaction_description:n {##1}
        \__chemmacros_begin_reaction:
        \__chemmacros_reaction_read:nnw {#2} {#4}
      }
      {
```

```
        \__chemmacros_end_reaction:
        \gdef\LWR@ThisAltText{}%                    lwarp
        \ignorespacesafterend
      }
  }

\cs_generate_variant:Nn \chemmacros_declare_reaction_env:nnnn {nnnV}

\RenewChemReaction {reaction}   {equation}
\RenewChemReaction {reaction*}  {equation*}
\RenewChemReaction {reactions}  {align}
\RenewChemReaction {reactions*} {align*}

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.15 Reactants

Recompiled for tabular ampersand processing, with the only change being `\StartDefiningTabulars`. `\xpatchcmd` does not work here.

```
\StartDefiningTabulars%     lwarp

% #1: star: include ID in table
\RenewDocumentCommand \printreactants {s}
  {
    \group_begin:
      \chemmacros_set_keys:nn {reactants} { switch = false }
      \int_step_variable:nNn
        { \seq_count:N \g_chemnum_initiated_compounds_seq }
        \l__chemmacros_reactants_tmpa_tl
        {
          \seq_put_right:Nx
            \l__chemmacros_reactants_tmpa_seq
            {
              \chemnum_cmpd:nnne { \c_false_bool } { \c_true_bool } {}
                {
                  \seq_item:NV
                    \g_chemnum_initiated_compounds_seq
                    \l__chemmacros_reactants_tmpa_tl
                }
              &
              \bool_if:nT {#1}
                {
                  \seq_item:NV
                    \g_chemnum_initiated_compounds_seq
                    \l__chemmacros_reactants_tmpa_tl
                  &
                }
              % TODO: expl3-command ??
              \solvent
                {
                  \seq_item:NV
                    \g_chemnum_initiated_compounds_seq
                    \l__chemmacros_reactants_tmpa_tl
                }
              \tabularnewline
            }
          \tl_set:Nx
            \l__chemmacros_reactants_tmpb_tl
```

```
            {
              \seq_item:NV
                \g_chemnum_initiated_compounds_seq
                \l__chemmacros_reactants_tmpa_tl
            }
          \chemmacros_reactants_list_subreactant:Vn
            \l__chemmacros_reactants_tmpb_tl
            {#1}
        }
      % TODO: longtable ?
      %       table customizable?
      % first draft of two styles
      \par
      \noindent
      \bool_if:NTF \l__chemmacros_reactants_printreactants_style_bool
        {
          \str_case:Vn \l__chemmacros_reactants_printreactants_style_str
            {
              {xltabular}
              {
                \chemmacros_if_package_loaded:nTF {xltabular}
                  {
                    \bool_if:nTF {#1}
                      {
                        \begin {xltabular}
                          { \textwidth }
                          { @{}ll>{\raggedright\arraybackslash}X@{} }
                      }
                      {
                        \begin {xltabular}
                          { \textwidth }
                          { @{}l>{\raggedright\arraybackslash}X@{} }
                      }
                    \seq_use:Nn \l__chemmacros_reactants_tmpa_seq { }
                    \end{xltabular}
                  }
                  {
                    \msg_expandable_error:nnnn
                      {chemmacros}
                      {package-not-loaded}
                      { \printreactants }
                      {xltabular}
                  }
              }
              {longtable}
              {
                \chemmacros_if_package_loaded:nTF {longtable}
                  {
                    \bool_if:nTF {#1}
                      {
                        \begin {longtable}[l]
                     { @{}ll>{\raggedright\arraybackslash}p{0.6\textwidth}@{} }
                      }
                      {
                        \begin {longtable}[l]
                     { @{}l>{\raggedright\arraybackslash}p{0.9\textwidth}@{} }
                      }
                    \seq_use:Nn \l__chemmacros_reactants_tmpa_seq { }
                    \end{longtable}
                  }
```

```
                {
                  \msg_expandable_error:nnnn
                    {chemmacros}
                    {package-not-loaded}
                    { \printreactants }
                    {longtable}
                }
            }
        }
    }
        {
            \msg_warning:nn {chemmacros} {missing-printreactants-style}
        }
    \group_end:
  }

% #1: full ID
% #2: star, include ID in table
\cs_gset_protected:Npn \chemmacros_reactants_list_subreactant:nn #1#2
  {
    \chemnum_if_subcompounds:nT {#1}
      {
        \int_step_variable:nNn
          { \chemnum_count_subcompounds:n {#1} }
          \l__chemmacros_reactants_tmpa_tl
          {
            \seq_put_right:Nx
            \l__chemmacros_reactants_tmpa_seq
              {
                \chemnum_cmpd:nnne { \c_false_bool } { \c_true_bool } {}
                  {
                    \exp_not:n {#1}
                    \exp_not:V \l_chemnum_compound_separator_tl
                    \chemnum_get_subcompound:nV
                      {#1}
                      \l__chemmacros_reactants_tmpa_tl
                  }
                &
                \bool_if:nT {#2}
                  {
                    #1
                    \l_chemnum_compound_separator_tl
                    \chemnum_get_subcompound:nV
                      {#1}
                      \l__chemmacros_reactants_tmpa_tl
                    &
                  }
                % TODO: expl3-command ??
                \solvent
                  {
                    #1
                    \l_chemnum_compound_separator_tl
                    \chemnum_get_subcompound:nV
                      {#1}
                      \l__chemmacros_reactants_tmpa_tl
                  }
                \tabularnewline
              }
          }
      }
```

```
  }
\cs_generate_variant:Nn \chemmacros_reactants_list_subreactant:nn {V}

\StopDefiningTabulars%      lwarp
```

§ 193.16 Redox

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{redox}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~redox}

\NewDocumentCommand \LWR@chemmacros@ox { s m >{\SplitArgument{1}{,}}m }
  {
    \IfBooleanTF {#1}
      { \chemmacros_ox:nnnn {#1} {#2} #3 }
      { \chemmacros_ox:nnnn {  } {#2} #3 }
  }

\RenewDocumentCommand \ox { s O{} m }
  {
    \begingroup
    \boolfalse{mathjax}
    \IfBooleanTF {#1}
      {
        \LWR@subsingledollar*{% yes hash
            \textbackslash{}%
            ox*%
            \{%
                \LWR@HTMLsanitizedetokenized{\detokenize{#3}}%
            \}% alt
        }{%
        star \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#2}}%
        }{%
            \LWR@chemmacros@ox* {#2} {#3}% contents
        }%
      }
      {
        \LWR@subsingledollar*{% yes hash
            \textbackslash{}%
            ox%
            \{%
                \LWR@HTMLsanitizedetokenized{\detokenize{#3}}%
            \}% alt
        }{%
          \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#2}}%
        }{%
            \LWR@chemmacros@ox {#2} {#3}% contents
        }%
      }
    \endgroup
  }

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.17 Scheme

Fix for chemmacros as of v5.8b, when using newfloat and babel:

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{scheme}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~scheme}

\ifdefstring{\schemename}{los}{
\SetupFloatingEnvironment{scheme}{
name = \chemmacros_translate:n {scheme-name}
}
}{}

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.18 Spectroscopy

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{spectroscopy}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~spectroscopy}

\cs_gset_protected:Npn \__chemmacros_nmr_base:nn #1#2
  {
    \group_begin:
      \tl_use:N \l__chemmacros_nmr_base_format_tl
      \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
        {
          \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl { \{ }
          \tl_put_right:Nn \g__chemmacros_nmr_element_coupled_tl { \} }
        }
      \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl {#2}
%       \chemmacros_formula:n { ^{#1} }
      \textsuperscript{#1}
      \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
        {
          \bool_if:NTF \l__chemmacros_nmr_parse_bool
            { \chemformula_ch:nV {} \g__chemmacros_nmr_element_coupled_tl }
            { \chemmacros_formula:V \g__chemmacros_nmr_element_coupled_tl }
        }
      \tl_use:N \l__chemmacros_nmr_element_method_connector_tl
      \tl_use:N \l__chemmacros_nmr_method_tl
    \group_end:
  }


\cs_gset_protected:Npn \chemmacros_nmr_position:n #1
  {
    \chemmacros_formula:x
      {
        \exp_not:V \g__chemmacros_nmr_element_tl
        \bool_if:NF \l__chemmacros_nmr_position_side_bool
          {
            \tl_if_eq:NnTF \l__chemmacros_nmr_position_tl {^}% lwarp
            { \textsuperscript{\exp_not:n { {#1} }} }% lwarp
            { \textsubscript{\exp_not:n { {#1} }} }% lwarp
%             \exp_not:V \l__chemmacros_nmr_position_tl
%             \exp_not:n { {#1} }
          }
      }
    \bool_if:NT \l__chemmacros_nmr_position_side_bool
      {
        \tl_use:N \l__chemmacros_nmr_position_tl
```

```
        \__chemmacros_nmr_position:n {#1}
      }
  }

\cs_gset_protected:Npn \__chemmacros_nmr_coupling:w (#1;#2)
  {
    \tl_set:Nn \l__chemmacros_nmr_coupling_bonds_tl
      {
        \l__chemmacros_nmr_coupling_bonds_pre_tl
        #1
        \l__chemmacros_nmr_coupling_bonds_post_tl
      }
    \bool_if:NTF \l__chemmacros_nmr_coupling_nuclei_sub_bool
      {
        \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
          {
%             \c_math_subscript_token
            \textsubscript% lwarp
              {
                \l__chemmacros_nmr_coupling_nuclei_pre_tl
                \chemmacros_formula:n {#2}
                \l__chemmacros_nmr_coupling_nuclei_post_tl
              }
          }
      }
      {
        \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
          {
            \l__chemmacros_nmr_coupling_nuclei_pre_tl
            \chemmacros_formula:n {#2}
            \l__chemmacros_nmr_coupling_nuclei_post_tl
          }
      }
    \__chemmacros_nmr_coupling_aux_i:w
  }
\AfterEndPreamble{% After \AtBeginDocument
% \NMR{<num>,<elem>}(<num>,<unit>)[<solvent>] ALL arguments are optional
% \NMR* same but without ": $\delta$" at end
\cs_gset_protected:Npn \chemmacros_nmr:nnnn #1#2#3#4
    {
    \bool_if:NT \l__chemmacros_nmr_list_bool { \item \scan_stop: }
    \group_begin:

        \mode_leave_vertical:

        \bool_set_false:N \l__chemmacros_nmr_frequency_bool
        \bool_set_false:N \l__chemmacros_nmr_solvent_bool
        \tl_if_empty:nF {#3}
        { \bool_set_true:N \l__chemmacros_nmr_frequency_bool }
        \tl_if_empty:nF {#4}
        { \bool_set_true:N \l__chemmacros_nmr_solvent_bool }
        \bool_if:nT
        {
            \l__chemmacros_nmr_frequency_bool
            ||
            \l__chemmacros_nmr_solvent_bool
        }
        { \bool_set_true:N \l__chemmacros_nmr_delimiters_bool }
        \bool_if:nT
        {
```

```
            \l__chemmacros_nmr_frequency_bool
            &&
            \l__chemmacros_nmr_solvent_bool
        }
        { \bool_set_true:N \l__chemmacros_nmr_comma_bool }
        \tl_if_empty:nTF {#2}
        {
            \__chemmacros_nmr_nucleus:VV
            \l__chemmacros_nmr_isotope_default_tl
            \l__chemmacros_nmr_element_default_tl
        }
        { \__chemmacros_nmr_nucleus:w #2 \q_stop }
        \mode_if_math:TF
        {
            \text
            {
                \group_begin:
                \tl_use:N \l__chemmacros_nmr_format_tl
\LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
                \__chemmacros_nmr_base:VV
                    \g__chemmacros_nmr_isotope_tl
                    \g__chemmacros_nmr_element_tl
                \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                    { ~ ( }
                \bool_if:NT \l__chemmacros_nmr_frequency_bool
                    { \__chemmacros_nmr_frequency:n {#3} }
                \bool_if:NT \l__chemmacros_nmr_comma_bool
                    { , ~ }
                \bool_if:NT \l__chemmacros_nmr_solvent_bool
                    { \chemmacros_formula:n {#4} }
                \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                    { ) }
                \tl_if_blank:nT {#1} {:~}
}}% lwarp
                \group_end:
            }
            \tl_if_blank:nT {#1}
            {
                \delta
                \text { \l__chemmacros_nmr_delta_tl }
                \bool_if:NT \l__chemmacros_nmr_use_equal_bool {=}
            }
        }
        {
            \group_begin:
            \tl_use:N \l__chemmacros_nmr_format_tl
\LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
            \__chemmacros_nmr_base:VV
                \g__chemmacros_nmr_isotope_tl
                \g__chemmacros_nmr_element_tl
            \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                {~(}
            \bool_if:NT \l__chemmacros_nmr_frequency_bool
                { \__chemmacros_nmr_frequency:n {#3} }
            \bool_if:NT \l__chemmacros_nmr_comma_bool
                {,~}
            \bool_if:NT \l__chemmacros_nmr_solvent_bool
                {
                \bool_if:NTF \l__chemmacros_nmr_parse_bool
```

```
%                     { \chemformula_ch:nn { } {#4} }%  original
                    {\ch{#4}}%  lwarp
                    {#4}
                }
            \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                {}}
}% lwarp
            \tl_if_blank:nT {#1} {:}
            \group_end:
            \tl_if_blank:nT {#1}
            {
                \tl_use:N \c_space_tl
                \c_math_toggle_token
                \delta
                \c_math_toggle_token
                \l__chemmacros_nmr_delta_tl
                \bool_if:NT \l__chemmacros_nmr_use_equal_bool {~=}
            }
        }
    \group_end:
    }
}% AfterEndPremble


\RenewDocumentCommand \chemmacros_data:w { smo }
  {
    \bool_if:NT \l__chemmacros_nmr_list_bool { \item }
      {
%         \tl_use:N \l__chemmacros_nmr_format_tl #2
        \tl_use:N \l__chemmacros_nmr_format_tl
        \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
            #2
            \IfNoValueF {#3} { ~ ( #3 ) }
        \IfBooleanT {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { : } }
        }}% lwarp
      }
   \IfBooleanF {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { ~ = } }
  }

}{}% Module loaded.
}% AtBeginDocument
```

§ 193.19 Thermodynamics

```
\AtBeginDocument{
\chemmacros_module_if_loaded:nTF{{thermodynamics}}{
\PackageInfo{lwarp}{Patching~chemmacros~module~thermodynamics}

\cs_gset_protected:Npn \chemmacros_state:nnnnnn #1#2#3#4#5#6
  {
    \group_begin:
      \chemmacros_set_keys:ne {thermodynamics}
        {
          \exp_not:n {#1} ,
          \tl_if_novalue:nF {#2} { subscript-left = \exp_not:n {#2} , }
          \tl_if_novalue:nF {#3} { superscript-left = \exp_not:n {#3} , }
          \tl_if_novalue:nF {#5} { subscript-right = \exp_not:n {#5} , }
          \tl_if_novalue:nF {#6} { superscript-right = \exp_not:n {#6} }
        }
        \LWR@subsingledollar*{% yes hashing
```

```
            \textbackslash{}state%
            \{\LWR@HTMLsanitizedetokenized{\detokenize{#4}}\}% alt
        }{%
            chemmacros_state% add'l hashing
            #1% options
            LSP \tl_use:N \l__chemmacros_state_sp_left_tl% super/subscripts
            LSB \tl_use:N \l__chemmacros_state_sb_left_tl
            RSP \tl_use:N \l__chemmacros_state_sp_right_tl
            RSB \tl_use:N \l__chemmacros_state_sb_right_tl
        }
        {
            \LWR@origensuredmath
                {
                \chemmacros_text:V \l__chemmacros_state_pre_tl
                \c_math_superscript_token
                    { \chemmacros_text:V \l__chemmacros_state_sp_left_tl }
```

Only add the subscripts if they are being used. This avoids causing an incorrect depth, as the empty subscript will be measured by TEX but cropped out by *pdfcrop*.

```
                \tl_if_empty:NTF \l__chemmacros_state_sb_left_tl
                {}
                {
                    \c_math_subscript_token
                      { \chemmacros_text:V \l__chemmacros_state_sb_left_tl }
                }
                #4
                \c_math_superscript_token
                    { \chemmacros_text:V \l__chemmacros_state_sp_right_tl }
                \tl_if_empty:NTF \l__chemmacros_state_sb_right_tl
                {}
                {
                    \c_math_subscript_token
                    { \chemmacros_text:V \l__chemmacros_state_sb_right_tl }
                }
                \chemmacros_text:V \l__chemmacros_state_post_tl
                }
        }
    \group_end:
  }
\cs_generate_variant:Nn \chemmacros_state:nnnnnn { nVVVVV }

\cs_gset_protected:Npn \chemmacros_declare_state:Nn #1#2
  {
    \chemmacros_define_keys:xn
      {thermodynamics/\chemmacros_remove_backslash:N #1}
      {
       pre              .meta:nn = {chemmacros/thermodynamics} { pre = ##1 } ,
       post             .meta:nn = {chemmacros/thermodynamics} { post = ##1 } ,
      superscript-left  .meta:nn = {chemmacros/thermodynamics} { superscript-left = ##1 } ,
      superscript-right .meta:nn = {chemmacros/thermodynamics} { superscript-right = ##1 } ,
        superscript       .meta:n  = { superscript-right = ##1 } ,
      subscript-left   .meta:nn = {chemmacros/thermodynamics} { subscript-left = ##1 } ,
      subscript-right  .meta:nn = {chemmacros/thermodynamics} { subscript-right = ##1 } ,
        subscript         .meta:n     = { subscript-left = ##1 } ,
        subscript-pos     .choices:nn =
          { left , right }
         { \tl_set_eq:NN \l__chemmacros_state_sb_pos_tl \l_keys_choice_tl } ,
        symbol            .tl_set:N = \l__chemmacros_state_symbol_tl ,
        unit              .tl_set:N = \l__chemmacros_state_unit_tl
```

```
      }
    \DeclareDocumentCommand #1 { sO{}D(){}m }
      {
        \group_begin:
          \chemmacros_set_keys:en
            {thermodynamics/\chemmacros_remove_backslash:N #1}
            {#2}
          \tl_if_blank:nF {##3}
            {
              \chemmacros_set_keys:ne {thermodynamics}
              { subscript-\l__chemmacros_state_sb_pos_tl = \exp_not:n {##3} }
            }
%           \LWR@origensuredmath
%             {
              \chemmacros_state:nVVVVV
                {##2}
                \c_novalue_tl
                \c_novalue_tl
                \l__chemmacros_state_symbol_tl
                \c_novalue_tl
                \c_novalue_tl
           \chemmacros_set_keys_groups:nnn {thermodynamics} {variables} {##2}
           \IfBooleanF {##1} { = \qty {##4} { \l__chemmacros_state_unit_tl } }
%             }
        \group_end:
      }
  }
```

The pre-existing macros are redefined with the new definition:

```
\RenewChemState \enthalpy { symbol = H , unit = \kilo\joule\per\mole }
\RenewChemState \entropy  { symbol = S , unit = \joule\per\kelvin\per\mole , pre = }
\RenewChemState \gibbs    { symbol = G , unit = \kilo\joule\per\mole }

}{}% Module loaded.
}% AtBeginDocument

\ExplSyntaxOff
```

File 85 **lwarp-chemnum.sty**

§ 194 Package

chemnum

(*Emulates or patches code by* Clemens Niederberger.)

chemnum (*Pkg*) chemnum is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{chemnum}[2016/04/14]

\ExplSyntaxOn

\cs_gset_protected:Npn \chemnum_compound_write:n #1
  {
    \chemnum_get_compound_property:nn {#1} {pre-main-label-code}
    \group_begin:
      \bool_if:NTF \l__chemnum_compound_local_bool
        { \l__chemnum_local_label_format_tl }
```

```
        { \chemnum_get_compound_property:nn {#1} {label-format} }
      {
        \LWR@textcurrentfont{
            \chemnum_get_compound_property:nn {#1} {counter-representation}
        }
      }
    \group_end:
    \chemnum_get_compound_property:nn {#1} {post-main-label-code}
  }

\cs_gset_protected:Npn \chemnum_subcompound_write:nn #1#2
  {
    \group_begin:
      \bool_if:NTF \l__chemnum_compound_local_bool
        { \l__chemnum_local_label_format_tl }
        { \chemnum_get_compound_property:nn {#1} {label-format} }
      {
        \LWR@textcurrentfont{
            \chemnum_get_subcompound_property:nnn {#1} {#2}
            {counter-representation}
        }
      }
    \group_end:
  }

\ExplSyntaxOff
```

File 86 **lwarp-chkfloat.sty**

§ 195 Package chkfloat

chkfloat (*Pkg*) chkfloat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{chkfloat}[2012/08/19]
```

File 87 **lwarp-chngpage.sty**

§ 196 Package chngpage

(*Emulates or patches code by* Peter Wilson.)

chngpage (*Pkg*) chngpage is ignored.

for HTML output: Discard all options for lwarp-chngpage:

```
\LWR@ProvidesPackageDrop{chngpage}[2009/10/20]
\LWR@origRequirePackage{lwarp-changepage}
```

File 88 **lwarp-cite.sty**

§ 197 Package cite

(*Emulates or patches code by* Donald Arseneau.)

cite (*Pkg*) cite is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{cite}[2015/02/27]
```

For the [super] option, the \kern must be removed:

```
\def\LWRCT@biblabel#1{\@citess{#1}\kern-\labelsep\,}

\ifdefstrequal{\@biblabel}{\LWRCT@biblabel}
{
    \def\@biblabel#1{\@citess{#1}}
}{}
```

For the [super] option, \textsuperscript is used instead of math superscript:

```
\def\@citess#1{\textsuperscript{#1}}

\DeclareDocumentCommand\citepunct{}{,\,\relax}
```

File 89 **lwarp-citeref.sty**

§ 198 Package citeref

(*Emulates or patches code by* Björn Briel.)

citeref (*Pkg*) citeref is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{citeref}[1999/27/05]

\def\@cprwrite#1={%
    \write\@auxout{\string\citepageref{#1}{\theLWR@previousautopagelabel}}%
}

\def\citepageref#1#2{%
    \xdef\cpr@testa{\@nameuse{cpr@last@#1}}%letzte Zitatstelle
    \xdef\cpr@testb{#2}% Seite dieser Zitatstelle
    \ifx\cpr@testa\cpr@testb%
        \relax% Konsekutive identische Seitenangaben weglassen
    \else%
        \@namexdef{cpr@last@#1}{#2}%
        \@ifundefined{cpr@#1}%
        {\@namexdef{cpr@#1}{\LWR@refwithsection{\BaseJobname-autopage-#2}}}%  lwarp
            {%      lwarp
                \@namexdef{cpr@#1}{\@nameuse{cpr@#1}, % space
                \LWR@refwithsection{\BaseJobname-autopage-#2}}%
            }%
    \fi
    }
```

File 90 **lwarp-CJK.sty**

§ 199 Package CJK

CJK (*Pkg*) CJK does not work with lwarp unless called from ctex.

for HTML output:

```
\IfPackageLoadedTF{xeCJK}{}{
    \LWR@loadnever{CJK}{ctex, xeCJK}
}

\LWR@ProvidesPackagePass{CJK}[2015/04/18]
```

File 91 **lwarp-CJKutf8.sty**

§ 200 Package

CJKutf8

CJKutf8 (*Pkg*) CJKutf8 does not work with lwarp unless called from ctex.

for HTML output:

```
\IfPackageLoadedTF{xeCJK}{}{
    \LWR@loadnever{CJKutf8}{ctex, xeCJK}
}

\LWR@ProvidesPackagePass{CJKutf8}[2015/04/18]
```

File 92 **lwarp-classicthesis.sty**

§ 201 Package

classicthesis

(*Emulates or patches code by* André Miede and Ivo Pletikosić.)

classicthesis (*Pkg*) classicthesis is emulated.

for HTML output: Discard all options for lwarp-classicthesis:

```
\LWR@ProvidesPackageDrop{classicthesis}[2018/06/03]

\RequirePackage{scrlayer-scrpage} % provides headers and footers (KOMA Script)
\RequirePackage{scrtime} % time access
\PassOptionsToPackage{titles}{tocloft}
\RequirePackage{textcase} % for \MakeTextUppercase
\RequirePackage[newparttoc]{titlesec} % newparttoc to write \part to .toc with \numberline
\RequirePackage{tocloft}
\PassOptionsToPackage{headinclude,footinclude}{typearea} % for classes other than KOMA
\RequirePackage{typearea}
\PassOptionsToPackage{marginal}{footmisc}% marginal flushmargin
\RequirePackage{footmisc}%
\RequirePackage{prelim2e}
\RequirePackage{remreset}%

\DeclareRobustCommand{\spacedallcaps}[1]{\textsc{\MakeTextUppercase{#1}}}
\DeclareRobustCommand{\spacedlowsmallcaps}[1]{\textsc{\MakeTextLowercase{#1}}}
\newcommand{\ctparttext}[1]{}
\newcommand{\tocEntry}[1]{}
\DeclareRobustCommand*{\deactivateaddvspace}{}%
\newlength{\beforebibskip}
```

File 93 **lwarp-cleveref.sty**

§ 202 Package

cleveref

(*Emulates or patches code by* Toby Cubitt.)

cleveref (*Pkg*) cleveref is patched for HTML, and limited MathJax emulation is added.

⚠ cleveref page numbers cleveref and varioref are supported, but printed page numbers do not map to HTML, so a section name or a text phrase are used for \cpageref and \cpagerefrange. This phrase includes \cpagerefFor, which defaults to "for".

Ex:

```
    \cpageref{tab:first,tab:second}
 in html becomes:
    "pages for table 4.1 and for table 4.2"
```

See \cpagerefFor at page 734 to redefine the message which is printed for page number references.

Table 16 on page 495 shows the data structure of the label/reference system as revised by lwarp and cleveref.

⚠ multiple labels For MathJax, each references is printed as an \eqref, without cleveref's description text. Page references are also printed as simple \eqrefs. Multiple labels in a single \cref will print as (???) in MathJax.

for HTML output:

```
1 \LWR@ProvidesPackagePass{cleveref}[2018/03/27]
```

The following patches are applied. Print-mode versions are not required since they all come down to \ref eventually, and \ref has a print-mode version.

\@@@setcref {⟨*kindofref*⟩} {⟨*label*⟩}

\@templabel becomes the section number.

```
2 \def\LWR@orig@@@setcref#1#2{\cref@getlabel{#2}{\@templabel}#1{\@templabel}{}{}}%
3
4 \ifdefequal{\@@setcref}{\LWR@orig@@@setcref}{% before v0.21
5     \renewcommand*{\@@setcref}[2]{#1{\ref{#2}}{}{}}
6 }{
7     \ifdefequal{\@@@setcref}{\LWR@orig@@@setcref}{% as of v0.21
8         \renewcommand*{\@@@setcref}[2]{%
9             #1{\ref{#2}}{}{}}
10    }{
11        \PackageWarningNoLine{lwarp-cleveref}{
12            Unknown version of cleveref.
13            \protect\cref\space will fail.
14        }%
15    }
16 }
```

\@@@setcrefrange {⟨*text*⟩} {⟨*label*⟩} {⟨*label*⟩}

```
\def\LWR@orig@@@setcrefrange#1#2#3{%
  \cref@getlabel{#2}{\@labela}%
  \cref@getlabel{#3}{\@labelb}%
  #1{\@labela}{\@labelb}{}{}{}{}}%

\ifdefequal{\@@setcrefrange}{\LWR@orig@@@setcrefrange}{
    \renewcommand{\@@setcrefrange}[3]{%
        #1{\ref{#2}}{\ref{#3}}{}{}{}{}%
    }
}{
    \ifdefequal{\@@@setcrefrange}{\LWR@orig@@@setcrefrange}{
        \renewcommand{\@@@setcrefrange}[3]{%
            #1{\ref{#2}}{\ref{#3}}{}{}{}{}%
        }
    }{
        \PackageWarningNoLine{lwarp-cleveref}{
            Unknown version of cleveref.
            \protect\crefrange\space will fail.
        }
    }
}
```

`\cpagerefFor` Redefinable word between “page(s)” and the page numbers.

```
\newcommand*{\cpagerefFor}{for}
```

`\@@@setcpageref` {⟨*typeofref*⟩} {⟨*label*⟩}, where typeofref is “page” or “pages”

```
\def\LWR@orig@@setcpageref#1#2{% before v0.21
  \cref@getpageref{#2}{\@temppage}#1{\@temppage}{}{}}%

\def\LWR@orig@@@setcpageref#1#2{% as of v0.21
  \cpageref@getlabel{#2}{\@temppage}#1{\@temppage}{}{}}%

\ifdefequal{\@@setcpageref}{\LWR@orig@@setcpageref}{
    \renewcommand*{\@@setcpageref}[2]{%
        #1{\cpagerefFor\ \cref{#2}}{}{}%
    }
}{
    \ifdefequal{\@@@setcpageref}{\LWR@orig@@@setcpageref}{
        \renewcommand*{\@@@setcpageref}[2]{%
            #1{\cpagerefFor\ \cref{#2}}{}{}%
        }
    }
    {
        \PackageWarningNoLine{lwarp-cleveref}{
            Unknown version of cleveref.
            \protect\cpageref\space will fail.
        }
    }
}

\def\LWR@orig@@setcpagerefrange#1#2#3{% before v0.21
  \cref@getpageref{#2}{\@pagea}%
  \cref@getpageref{#3}{\@pageb}%
  #1{\@pagea}{\@pageb}{}{}{}{}%

```

```
\def\LWR@orig@@@setcpagerefrange#1#2#3{% as of v0.21
  \cpageref@getlabel{#2}{\@pagea}%
  \cpageref@getlabel{#3}{\@pageb}%
  #1{\@pagea}{\@pageb}{}{}{}{}}%

\ifdefequal{\@@setcpagerefrange}{\LWR@orig@@setcpagerefrange}{
    \renewcommand*{\@@setcpagerefrange}[3]{%
        #1{\cpagerefFor\ \cref{#2}}{\cref{#3}}{}{}{}{}%
    }
}{
    \ifdefequal{\@@@setcpagerefrange}{\LWR@orig@@@setcpagerefrange}{
        \renewcommand*{\@@@setcpagerefrange}[3]{%
            #1{\cpagerefFor\ \cref{#2}}{\cref{#3}}{}{}{}{}%
        }
    }
    {
        \PackageWarningNoLine{lwarp-cleveref}{
            Unknown version of cleveref.
            \protect\cpagerefrange\space will fail.
        }
    }
}
```

If hyperref is loaded, cleveref defines starred versions of the following, but since hyperref is only emulated, starred versions are defined here:

```
\LWR@absorbstar{cref}
\LWR@absorbstar{Cref}
\LWR@absorbstar{crefrange}
\LWR@absorbstar{Crefrange}
\LWR@absorbstar{cpageref}
\LWR@absorbstar{Cpageref}
\LWR@absorbstar{cpagerefrange}
\LWR@absorbstar{Cpagerefrange}
\LWR@absorbstar{labelcref}
\LWR@absorbstar{labelcpageref}
```

If hyperref is loaded, cleveref also defines starred versions of varioref macros, so they are defined here.

```
\IfPackageLoadedTF{varioref}{
    \LWR@absorbstar{vref}
    \LWR@absorbstar{Vref}
    \LWR@absorbstar{vrefrange}
    \LWR@absorbstar{Vrefrange}
    \LWR@absorbstar{fullref}
    \LWR@absorbstar{Fullref}
}{}% varioref


\IfClassLoadedTF{memoir}{
\AtBeginDocument{
\def\sf@memsub@label(#1)#2{%
  \protected@edef\mem@currentlabelname{#1}%
  \sf@@memsub@label{#2}}
}
}{}


\IfPackageLoadedTF{subfig}{
```

```
\def\sf@sub@label(#1)#2{%
  \ifhyperrefloaded
    \protected@edef\@currentlabelname{%
      \expandafter\strip@period #1\relax.\relax\@@@}%
  \fi
  \sf@@sub@label{#2}}
}{}
```

File 94 **lwarp-clrdblpg.sty**

§ 203 Package clrdblpg

clrdblpg (*Pkg*) clrdblpg is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{clrdblpg}[2018/04/21]
```

File 95 **lwarp-cmbright.sty**

§ 204 Package cmbright

(*Emulates or patches code by* Walter Schmidt.)

cmbright (*Pkg*) cmbright is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options, except `slantedGreek` is honored, and `\mathbold` is available.

The dedicated macros for upright Greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{cmbright}[2005/04/13]

\LWR@infoprocessingmathjax{cmbright}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

\IfPackageLoadedWithOptionsTF{cmbright}{slantedGreek}
{
    \LWR@mathjax@addgreek@u@it*{}{}
}
{}

\LWR@mathjax@addgreek@u@up*{up}{}

\CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}

\end{warpMathJax}
```

File 96 **lwarp-cmdtrack.sty**

§ 205 Package cmdtrack

cmdtrack (*Pkg*) cmdtrack is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{cmdtrack}[2012/12/18]
\newcommand{\untrack}[1]{}
```

File 97 **lwarp-colonequals.sty**

§ 206 Package colonequals

(*Emulates or patches code by* Heiko Oberdiek.)

colonequals (*Pkg*) colonequals is used as-is for SVG math, and is emulated for MathJax.

Since Unicode symbols are not available for each of the following, only two are used for the single and double colons, and the other symbols are derived in a consistent manner. Occasional negative space is added as well. This may need to be undone for some fonts.

for HTML output:

```
\LWR@ProvidesPackagePass{colonequals}[2016/05/16]
\begin{warpMathJax}
\LWR@infoprocessingmathjax{colonequals}
\CustomizeMathJax{\newcommand{\ratio}{\mathrel{\unicode{x2236}}}}
\CustomizeMathJax{\newcommand{\coloncolon}{\mathrel{\unicode{x2237}}}}
\CustomizeMathJax{\newcommand{\colonequals}{\mathrel{\unicode{x2236}\!=}}}
\CustomizeMathJax{\newcommand{\coloncolonequals}{\mathrel{\unicode{x2237}\!=}}}
\CustomizeMathJax{\newcommand{\equalscolon}{\mathrel{=\!\unicode{x2236}}}}
\CustomizeMathJax{\newcommand{\equalscoloncolon}{\mathrel{=\!\unicode{x2237}}}}
\CustomizeMathJax{\newcommand{\colonminus}{\mathrel{\unicode{x2236}-}}}
\CustomizeMathJax{\newcommand{\coloncolonminus}{\mathrel{\unicode{x2237}-}}}
\CustomizeMathJax{\newcommand{\minuscolon}{\mathrel{-\unicode{x2236}}}}
\CustomizeMathJax{\newcommand{\minuscoloncolon}{\mathrel{-\unicode{x2237}}}}
\CustomizeMathJax{\newcommand{\colonapprox}{\mathrel{\unicode{x2236}\!\approx}}}
\CustomizeMathJax{\newcommand{\coloncolonapprox}{\mathrel{\unicode{x2237}\!\approx}}}
\CustomizeMathJax{\newcommand{\approxcolon}{\mathrel{\approx\!\unicode{x2236}}}}
\CustomizeMathJax{\newcommand{\approxcoloncolon}{\mathrel{\approx\!\unicode{x2237}}}}
\CustomizeMathJax{\newcommand{\colonsim}{\mathrel{\unicode{x2236}\!\sim}}}
\CustomizeMathJax{\newcommand{\coloncolonsim}{\mathrel{\unicode{x2237}\!\sim}}}
\CustomizeMathJax{\newcommand{\simcolon}{\mathrel{\sim\!\unicode{x2236}}}}
\CustomizeMathJax{\newcommand{\simcoloncolon}{\mathrel{\sim\!\unicode{x2237}}}}
\end{warpMathJax}
```

File 98 **lwarp-color.sty**

§ 207 Package
# color

color (*Pkg*) Allowed but ignored. xcolor is then required as well.

color is superceded by xcolor, and lwarp requires several of the features of xcolor. When color is requested, xcolor is loaded as well.

for HTML output:
```
\LWR@ProvidesPackageDrop{color}[2016/07/10]
\RequirePackage{xcolor}
```

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

```
\let\color@endgroup\endgroup
```

File 99 **lwarp-colortbl.sty**

§ 208 Package
# colortbl

colortbl (*Pkg*) colortbl is used as-is for print output, and emulated for HTML.

⚠ row/cell color Only use \rowcolor and \cellcolor at the start of a row, in that order.

colortbl ignores the overhang arguments.

for HTML output: A placeholder definition is forgotten first:

```
\let\rowcolor\relax

\LWR@ProvidesPackagePass{colortbl}[2018/12/12]
```

The following \LWR@HTML versions are used inside an HTML tabular.

\columncolor `[`⟨*model*⟩`]` `{`⟨*color*⟩`}` `[`⟨*left overhang*⟩`]` `[`⟨*right overhang*⟩`]`

\LWR@getmynexttoken is not used here because \columncolor is not used inside the data area of the tabular.

\columncolor is provided here to satisfy \LWR@formatted's test for the existence of the print-mode macro.

```
\ProvideDocumentCommand{\columncolor}{O{named} m o o}{}%

\NewDocumentCommand{\LWR@HTML@columncolor}{O{named} m o o}{%
    \convertcolorspec{#1}{#2}{HTML}\LWR@columnHTMLcolor%
    \LWR@addtabularcellcolor%
}

\AtBeginDocument{\LWR@formatted{columncolor}}
```

\LWR@getmynexttoken is used for \rowcolor because it is used inside the data area of the tabular.

\rowcolor [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩]

```
\NewDocumentCommand{\LWR@HTML@rowcolor}{O{named} m o o}{%
    \convertcolorspec{#1}{#2}{HTML}\LWR@rowHTMLcolor%
    \LWR@getmynexttoken%
}

\AtBeginDocument{\LWR@expandableformatted{rowcolor}}
```

\cellcolor [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩]

```
\NewDocumentCommand{\LWR@HTML@cellcolor}{O{named} m o o}{%
    \convertcolorspec{#1}{#2}{HTML}\LWR@cellHTMLcolor%
    \LWR@addtabularcellcolor%
}

\AtBeginDocument{\LWR@formatted{cellcolor}}
```

\arrayrulecolor [⟨*model*⟩] {⟨*color*⟩}

The HTML version for use outside a `tabular`. Inside a `tabular`, `\LWR@HTML@arrayrulecolornexttoken` is used instead.

```
\newcommand{\LWR@HTML@arrayrulecolor}[2][named]{%
    \convertcolorspec{#1}{#2}{HTML}\LWR@ruleHTMLcolor%
}

\AtBeginDocument{\LWR@expandableformatted{arrayrulecolor}}
```

\LWR@arrayrulecolornexttoken [⟨*model*⟩] {⟨*color*⟩}

The HTML version for use inside a `tabular`.

```
\newcommand{\LWR@HTML@arrayrulecolornexttoken}[2][named]{%
    \convertcolorspec{#1}{#2}{HTML}\LWR@ruleHTMLcolor%
    \LWR@getmynexttoken%
}

\AtBeginDocument{\LWR@expandableformatted{arrayrulecolornexttoken}}
```

\doublerulesepcolor [⟨*model*⟩] {⟨*color*⟩}

The version for use outside a `tabular`.

```
\newcommand{\LWR@HTML@doublerulesepcolor}[2][named]{}

\AtBeginDocument{\LWR@expandableformatted{doublerulesepcolor}}
```

\LWR@doublerulesepcolornexttoken [⟨*model*⟩] {⟨*color*⟩}

The version for use inside a `tabular`.

```
\newcommand{\LWR@HTML@doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}

\AtBeginDocument{\LWR@expandableformatted{doublerulesepcolornexttoken}}
```

For MathJax, use the MathJax package. The unused macro options are ignored.

```
\begin{warpMathJax}
```

```
42
43 \CustomizeMathJax{\require{colortbl}}
44 \CustomizeMathJax{\let\LWRorigcolumncolor\columncolor}
45 \CustomizeMathJax{\renewcommand{\columncolor}[2][named]{%
46     \LWRorigcolumncolor[#1]{#2}%
47     \LWRabsorbtwooptions%
48 }}
49
50 \CustomizeMathJax{\let\LWRorigrowcolor\rowcolor}
51 \CustomizeMathJax{\renewcommand{\rowcolor}[2][named]{%
52     \LWRorigrowcolor[#1]{#2}%
53     \LWRabsorbtwooptions%
54 }}
55
56 \CustomizeMathJax{\let\LWRorigcellcolor\cellcolor}
57 \CustomizeMathJax{\renewcommand{\cellcolor}[2][named]{%
58     \LWRorigcellcolor[#1]{#2}%
59     \LWRabsorbtwooptions%
60 }}
61
62 \end{warpMathJax}
```

File 100 **lwarp-continue.sty**

§ 209 Package continue

continue (*Pkg*) continue is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{continue}}[2018/12/09]

2 \newcommand*{\flagcont}{}
3 \newcommand*{\flagend}{}
4 \newcommand*{\flagword}{}
5 \newcommand*{\preflagword}{}
6 \newcommand*{\postflagword}{}
7 \newlength\contsep
8 \newlength\contdrop
```

File 101 **lwarp-copyrightbox.sty**

§ 210 Package copyrightbox

(*Emulates or patches code by* Thomas Fischer, Ives van der Flaas.)

copyrightbox (*Pkg*) copyrightbox is emulated for use by lwarp.

The entire copyright box is placed inside a of class copyrightbox.

The contents are placed inside a of class copyrightboxcontents.

The copyright notice is placed inside a of class copyrightboxnote.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{copyrightbox}[2011/11/27]
```

```
\newcommand{\copyrightbox}[3][r]{%
\begin{BlockClass}[
    display: inline-flex;
    flex-direction: column ;
]{copyrightbox}
\begin{BlockClass}{copyrightboxcontents}
#2
\end{BlockClass}
\begin{BlockClass}{copyrightboxnote}
#3
\end{BlockClass}
\end{BlockClass}
}

\newcommand{\CRB@setcopyrightfont}{}
\newcommand{\CRB@setcopyrightparagraphstyle}{}
```

File 102 **lwarp-crop.sty**

§ 211 Package

crop

(*Emulates or patches code by* Melchior FRANZ.)

crop (*Pkg*) crop is ignored.

for HTML output: Discard all options for lwarp-crop:

```
\LWR@ProvidesPackageDrop{crop}[2003/05/20]
```

```
\newcommand*{\crop}[1][]{}
\newcommand*{\cropdef}[6][]{}
```

File 103 **lwarp-ctable.sty**

§ 212 Package

ctable

(*Emulates or patches code by* Wybo Dekker.)

ctable (*Pkg*) ctable is patched for use by lwarp.

⚠ Misplaced alignment tab character &

Use `\StartDefiningTabulars` before one or more `\ctables`, and `\StopDefiningTabulars` after. These change the meaning of the ampersand & character.

for HTML output:

```
\LWR@ProvidesPackagePass{ctable}[2015/10/17]
```

The following is in the original:

```
\newcommand{\LWR@HTML@ctable}[4][]{%
   \let\@CTtaborfig  \@dfltCTtaborfig
   \let\@CTalign     \@dfltCTalign
   \let\@CTsideways  \@dfltCTsideways
   \let\@CTcontinued \empty
   \let\@CTpos       \@dfltCTpos
   \let\@CTcaption   \empty
```

```
\let\@CTcap       \undefined
\let\@CTlabel     \empty
\let\@CTbotcap    \@dfltCTbotcap
\let\@CTstarred   \@dfltCTstarred
\let\@CTsuper     \@dfltCTsuper
\let\@CTnotespar  \@dfltCTnotespar
\let\@CTdoinside  \@dfltCTdoinside
\let\@CTbgopacity \@dfltCTbgopacity
\@CTframerule     \@dfltCTframerule
\@CTcaptionskip   \@dfltCTcaptionskip
\@CTframesep      \@dfltCTframesep
\@CTwidth         \@dfltCTwidth
\@CTmaxwidth      \@dfltCTmaxwidth
\@CTmincapwidth   \@dfltCTmincapwidth
\@CTfooterwidth   \@dfltCTfooterwidth
\def\@CTfgactual {@dfltCTframefg}%
\def\@CTbgactual {@dfltCTframebg}%
\def\@CTbeg      {\begin{\@CTsideways\@CTtaborfig\@CTstarred}}%
\def\@CTbegin    {\@CTbeg}%
\def\@CTend      {\end{\@CTsideways\@CTtaborfig\@CTstarred}}%
\setkeys{CT}{#1}%
\ifx\@CTcap\undefined\let\@CTcap\@CTcaption\fi
\ifx\@CTcap\empty
  \if@CTcaptionloaded\else
    \PackageWarningNoLine{lwarp-ctable}{\MessageBreak
       An empty cap= option prevents lot/loc entry only\MessageBreak
       if the caption package is loaded!}
  \fi
\fi
\if@CTinmemoir\else
   \ifx\@CTbotcap\undefined
      \PackageError{lwarp-ctable}{\MessageBreak
        You can, currently, use the sidecap option only with\MessageBreak
         memoir documents. Use topcap or botcap only}
         {}
   \fi
\fi
\ifdim\@CTwidth=0pt\else
   \ifdim\@CTmaxwidth=0pt\else
      \PackageError{lwarp-ctable}{\MessageBreak
       You may not use the width and maxwidth options together\MessageBreak
          Use either width or maxwidth}
          {}
   \fi
\fi
\ifx\@CTpos\empty
   \ifx\@CTsideways\empty\else
   \PackageError{lwarp-ctable}{\MessageBreak
      You may not use the pos and sideways options together\MessageBreak
      Rotated tables and figures are always typeset on a separate page}
     {}
   \fi
\fi
\ifx\@CTcaption\empty
   \ifx\@CTlabel\empty\else
      \PackageError{lwarp-ctable}{\MessageBreak
         You may not label a captionless table\MessageBreak
         Such a label can't be referenced}
         {}
   \fi
```

```
  \fi
```

Some of the original, regarding computing the width of `\CT@t`, is removed here.

```
  \@CTbegin
     \ifx\@CTcontinued\empty\else\addtocounter{\@CTtaborfig}{-1}\fi
     \@CTalign
```

lwarp's patches begin here:

```
   \begin{center}
       \setlength{\fboxrule}{\@CTframerule}
       \setlength{\fboxsep}{\@CTframesep}
       \LWR@forceminwidth{\fboxrule}%  lwarp
       \convertcolorspec{named}{\@CTbgactual}{HTML}\LWR@tempcolor% lwarp
       \begin{BlockClass}[%                              lwarp
           border:
               \LWR@printlength{\LWR@atleastonept}
               solid
               \LWR@colorstyle{named}{\@CTfgactual} ; %
           padding:\LWR@printlength{\fboxsep} ; %
           \ifdefstring{\LWR@tempcolor}{FFFFFF}{}{%
               background: \LWR@colorstyle{named}{\@CTbgactual} ; %
           }%
       ]{fminipage}%       lwarp
        \ifx\@CTbotcap\@CTfalse\@CTcaption\vskip\@CTcaptionskip\fi
        \ifx\@CTbotcap\undefined%
            \begin{sidecaption}[\@CTcap]{\@CTcaption}[\@CTlabel]
        \fi
        \@CTdoinside
        \begin{tabularx}{\linewidth}{#2}%      lwarp
           #4%
        \end{tabularx}%                    lwarp
        \def\@CTfootnotes{#3}%
        \ifx#3\empty\else{% append footnotes, if any
           \begin{BlockClass}{tnotes}%     lwarp
           #3
           \end{BlockClass}%               lwarp
        }
        \fi
        \ifx\@CTbotcap\undefined\end{sidecaption}\fi
        \ifx\@CTbotcap\@CTtrue\vskip\@CTcaptionskip\@CTcaption\fi
       \end{BlockClass}
    \end{center}
   \@CTend
}
\LWR@formatted{ctable}
```

Required to properly detect the toprule:

```
\LetLtxMacro\FL\toprule
```

Table notes are redefined for HTML:

```
\newcommand{\LWR@HTML@tmark}[1][a]{%
   \textsuperscript{\textrm{\textit{#1}}}
}
\LWR@formatted{tmark}

```

```
\newcommand{\LWR@HTML@tnote}[2][a]{%
    \tmark[#1]\,#2\par
}
\LWR@formatted{tnote}
```

File 104 **lwarp-cuted.sty**

§ 213 Package cuted

(*Emulates or patches code by* Sigitas Tolušis.)

cuted (*Pkg*) cuted is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{cuted}[2021/10/04]

\newenvironment{strip}{}{}
\newskip\stripsep
\newtoks\preCutedStrip \preCutedStrip{}
\newtoks\postCutedStrip \postCutedStrip{}
\def\oldcolsbreak#1{}
```

File 105 **lwarp-cutwin.sty**

§ 214 Package cutwin

(*Emulates or patches code by* Peter Wilson and Alan Hoenig.)

cutwin (*Pkg*) cutwin is emulated.

for HTML output: Discard all options for lwarp-cutwin:

```
\LWR@ProvidesPackageDrop{cutwin}[2010/09/29]

\newcommand*{\opencutleft}{}
\newcommand*{\opencutright}{}
\newcommand*{\opencutcenter}{}
\newcommand*{\cutfuzz}{}

\newenvironment{cutout}[4]
{\marginpar{\windowpagestuff}}
{}

\newcommand*{\windowpagestuff}{}

\newcommand*{\pageinwindow}{%
% \begin{minipage}{.3\linewidth}
\windowpagestuff
% \end{minipage}
}

\newenvironment{shapedcutout}[3]
{\marginpar{\picinwindow}}
{}

```

```
\newcommand*{\putstuffinpic}{}

\newcommand*{\picinwindow}{%
\begin{picture}(0,0)
\putstuffinpic
\end{picture}}
```

File 106 **lwarp-dblfloatfix.sty**

§ 215 Package dblfloatfix

dblfloatfix (*Pkg*) dblfloatfix is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dblfloatfix}[2012/12/31]
```

File 107 **lwarp-dblfnote.sty**

§ 216 Package dblfnote

(*Emulates or patches code by* Hiroshi Nakashima.)

dblfnote (*Pkg*) dblfnote is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dblfnote}[1999/07/14]

\newcounter{DFNsloppiness}
\newdimen\DFNcolumnsep
\newdimen\DFNcolumnwidth
\def\DFNallowcbreak{}
\def\DFNinhibitcbreak{}
\def\DFNtrysingle{}
\def\DFNalwaysdouble{}
\def\DFNruleboth{}
\def\DFNruleleft{}
```

File 108 **lwarp-dcolumn.sty**

§ 217 Package dcolumn

dcolumn (*Pkg*) dcolumn is used as-is in a `lateximage`, and is emulated by the lwarp core.

dcolumn used to be `\LWR@ProvidesPackageDrop` in prior versions of lwarp, but is now supported for print mode.

```
\LWR@ProvidesPackagePass{dcolumn}[2014/10/28]
```

Due to how the `D` column is created, cannot use `\HTMLnewcolumntype` here. An HTML version neutralizes the lower-level macros, leaving a `c` column type.

```
\newcommand*{\LWR@HTML@DC@}[3]{}
```

```
\LWR@formatted{DC@}

\providecommand*{\DC@end}{}

\newcommand*{\LWR@HTML@DC@end}{}
\LWR@formatted{DC@end}
```

File 109 **lwarp-decimal.sty**

§ 218 Package decimal

(*Emulates or patches code by* A. Syropoulos and R. W. D. Nickalls.)

decimal (*Pkg*) decimal works as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{decimal}[2011/06/03]

\begin{warpMathJax}
\CustomizeMathJax{\def\.{\mbox{.}}}
\end{warpMathJax}
```

File 110 **lwarp-decorule.sty**

§ 219 Package decorule

(*Emulates or patches code by* Peter Flynn.)

decorule (*Pkg*) decorule is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{decorule}[2020/04/01]

\xpretocmd{\decorule}
    {\begin{lateximage}*[decorule]}
    {}
    {\LWR@patcherror{decorule}{decorule A}}

\xapptocmd{\decorule}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{decorule}{decorule B}}
```

File 111 **lwarp-diagbox.sty**

§ 220 Package diagbox

(*Emulates or patches code by* Leo Liu.)

diagbox (*Pkg*) diagbox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{diagbox}[2016/12/28]
```

To restore print-mode inside a `lateximage`:

```
\LetLtxMacro\LWR@origdiagbox@double\diagbox@double
\LetLtxMacro\LWR@origdiagbox@triple\diagbox@triple

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\diagbox@double\LWR@origdiagbox@double%
\LetLtxMacro\diagbox@triple\LWR@origdiagbox@triple%
}
```

`\LWR@diagbox@AB` {⟨*E/W*⟩} {⟨*A*⟩} {⟨*E/W*⟩} {⟨*B*⟩}

```
\newcommand{\LWR@diagbox@AB}[4]{
\begingroup%
\LetLtxMacro\\\newline%
\BlockClassSingle{diagbox#1}{#2}%
\BlockClassSingle{diagbox#3}{#4}%
\endgroup%
\LWR@stoppars%
}
```

`\LWR@diagboxNW` {⟨*A*⟩} {⟨*B*⟩}

```
\newcommand{\LWR@diagboxNW}[2]{%
\LWR@diagbox@AB{E}{#2}{W}{#1}%
}
```

Likewise for NE, SW, SE:

```
\newcommand{\LWR@diagboxNE}[2]{%
\LWR@diagbox@AB{W}{#1}{E}{#2}%
}

\let\LWR@diagboxSW\LWR@diagboxNE
\let\LWR@diagboxSE\LWR@diagboxNW
```

`\diagbox@double` {⟨*keys*⟩} {⟨*A*⟩} {⟨*B*⟩}

```
\def\diagbox@double#1#2#3{%
\setkeys{diagbox}{dir=NW,#1}%
\@nameuse{LWR@diagbox\diagbox@dir}{#2}{#3}%
}
```

`\LWR@diagboxTNW` {⟨*title*⟩} {⟨*A*⟩} {⟨*B*⟩}

```
\newcommand{\LWR@diagboxTNW}[3]{%
\BlockClassSingle{diagboxtitleN}{#1}
\LWR@diagboxNW{#2}{#3}
}
```

Likewise for NE, SW, SE:

```
\newcommand{\LWR@diagboxTNE}[3]{%
\BlockClassSingle{diagboxtitleN}{#1}
\LWR@diagboxNE{#2}{#3}
}

\newcommand{\LWR@diagboxTSW}[3]{%
```

```
\LWR@diagboxSW{#2}{#3}
\BlockClassSingle{diagboxtitleS}{#1}
\LWR@stoppars%
}

\newcommand{\LWR@diagboxTSE}[3]{%
\LWR@diagboxSE{#2}{#3}
\BlockClassSingle{diagboxtitleS}{#1}
\LWR@stoppars%
}
```

\diagbox@triple {⟨*keys*⟩} {⟨*A*⟩} {⟨*T*⟩} {⟨*B*⟩}

```
\def\diagbox@triple#1#2#3#4{%
\setkeys{diagbox}{dir=NW,#1}%
\@nameuse{LWR@diagboxT\diagbox@dir}{#3}{#2}{#4}%
}
```

File 112 **lwarp-dingbat.sty**

§ 221 Package dingbat

(*Emulates or patches code by* Scott Pakin.)

dingbat (*Pkg*) dingbat is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{dingbat}[2001/04/27]

\newcommand*{\LWR@dingbatsymbol}[1]{\HTMLunicode{#1}}

\newcommand{\LWR@HTML@rightpointright}{\LWR@dingbatsymbol{261E}}
\newcommand{\LWR@HTML@leftpointright}{\LWR@dingbatsymbol{261E}}
\newcommand{\LWR@HTML@leftthumbsdown}{\LWR@dingbatsymbol{1F44E}}
\newcommand{\LWR@HTML@leftthumbsup}{\LWR@dingbatsymbol{1F44D}}
\newcommand{\LWR@HTML@rightpointleft}{\LWR@dingbatsymbol{261C}}
\newcommand{\LWR@HTML@rightthumbsdown}{\LWR@dingbatsymbol{1F44E}}
\newcommand{\LWR@HTML@rightthumbsup}{\LWR@dingbatsymbol{1F44D}}
\newcommand{\LWR@HTML@squarewithdots}{\LWR@dingbatsymbol{25C7}}
\newcommand{\LWR@HTML@filledsquarewithdots}{\LWR@dingbatsymbol{25C6}}
\newcommand{\LWR@HTML@Sborder}{\LWR@dingbatsymbol{271A}}
\newcommand{\LWR@HTML@Zborder}{\LWR@dingbatsymbol{274B}}
\newcommand{\LWR@HTML@largepencil}{\LWR@dingbatsymbol{270E}}
\newcommand{\LWR@HTML@anchor}{\LWR@dingbatsymbol{2693}}
\newcommand{\LWR@HTML@carriagereturn}{\LWR@dingbatsymbol{23CE}}
\newcommand{\LWR@HTML@checkmark}{\LWR@dingbatsymbol{2713}}
\newcommand{\LWR@HTML@eye}{\LWR@dingbatsymbol{1F441}}
\newcommand{\LWR@HTML@satellitedish}{\LWR@dingbatsymbol{1F4E1}}
\newcommand{\LWR@HTML@smallpencil}{\LWR@dingbatsymbol{270E}}

\LWR@formatted{rightpointright}
\LWR@formatted{leftpointright}
\LWR@formatted{leftthumbsdown}
\LWR@formatted{leftthumbsup}
\LWR@formatted{rightpointleft}
\LWR@formatted{rightthumbsdown}
\LWR@formatted{rightthumbsup}
\LWR@formatted{squarewithdots}
```

```
\LWR@formatted{filledsquarewithdots}
\LWR@formatted{Sborder}
\LWR@formatted{Zborder}
\LWR@formatted{largepencil}
\LWR@formatted{anchor}
\LWR@formatted{carriagereturn}
\LWR@formatted{checkmark}
\LWR@formatted{eye}
\LWR@formatted{satellitedish}
\LWR@formatted{smallpencil}
```

File 113 **lwarp-DotArrow.sty**

§ 222 Package

DotArrow

(*Emulates or patches code by* Sven Schneider.)

DotArrow (*Pkg*) DotArrow is patched for use by lwarp, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{DotArrow}[2007/02/12]
```

The width must be recomputed each time, depending on print or HTML output.

```
\xpretocmd{\dotarrow}{\settowidth{\oneWidth}{\onePartX}}{}{}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\dotarrow}[1]{\stackrel{#1}{\unicode{x21E2}}}}
\end{warpMathJax}
```

File 114 **lwarp-dotlessi.sty**

§ 223 Package

dotlessi

(*Emulates or patches code by* Javier Bezos.)

dotlessi (*Pkg*) dotlessi is used as-is for SVG math, and is emulated for MathJax.

⚠ HTML \dotlessj Use \usepackage{cmap} if \dotlessj does not appear in HTML in text mode. See section 7.4.

⚠ not bold For MathJax, use \boldsymbol instead of \mathbf.

for HTML output:

```
\LWR@ProvidesPackagePass{dotlessi}[1999/10/12]
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\let\dotlessi\imath}
\CustomizeMathJax{\let\dotlessj\jmath}
\end{warpMathJax}
```

File 115 **lwarp-dprogress.sty**

§ 224 Package dprogress

dprogress (*Pkg*) dprogress is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dprogress}[2008/02/21]
```

File 116 **lwarp-draftcopy.sty**

§ 225 Package draftcopy

draftcopy (*Pkg*) draftcopy is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{draftcopy}[2002/02/25]

\newcommand{\draftcopyVersion}[1]{}
\newcommand{\draftcopySetGrey}[1]{}
\newcommand{\draftcopySetScale}[1]{}
\newcommand{\draftcopySetScaleFactor}[1]{}
\newcommand{\draftcopyFirstPage}[1]{}
\newcommand{\draftcopyLastPage}[1]{}
\newcommand{\draftcopyName}[2]{}
\newcommand{\draftcopyPageTransform}[1]{}
\newcommand{\draftcopyBottomTransform}[1]{}
\newcommand{\draftcopyPageX}[1]{}
\newcommand{\draftcopyPageY}[1]{}
\newcommand{\draftcopyBottomX}[1]{}
\newcommand{\draftcopyBottomY}[1]{}
```

File 117 **lwarp-draftfigure.sty**

§ 226 Package draftfigure

draftfigure (*Pkg*) draftfigure is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{draftfigure}[2017/07/19]
\RequirePackage{xkeyval}

\define@key{draftfigure}{code}{}
\define@key{draftfigure}{noframe}[true]{}
\define@key{draftfigure}{filename}[true]{}
\define@key{draftfigure}{content}[]{}
\define@key{draftfigure}{style}[normal]{}
\define@key{draftfigure}{position}[left]{}
\define@key{draftfigure}{size}[normal]{}
\newcommand\setdf[1]{\setkeys{draftfigure}{#1}}
```

File 118 **lwarp-draftwatermark.sty**

§ 227 Package draftwatermark

(*Emulates or patches code by* Sergio Callegari.)

draftwatermark (*Pkg*) draftwatermark is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{draftwatermark}[2020/03/14]

\newcommand{\DraftwatermarkOptions}[1]{}
\newcommand{\DraftwatermarkStdMark}{}
\newcommand{\SetWatermarkAngle}[1]{}
\newcommand{\SetWatermarkColor}[1]{}
\newcommand{\SetWatermarkLightness}[1]{}
\newcommand{\SetWatermarkFontSize}[1]{}
\newcommand{\SetWatermarkScale}[1]{}
\newcommand{\SetWatermarkHorCenter}[1]{}
\newcommand{\SetWatermarkVertCenter}[1]{}
\newcommand{\SetWatermarkText}[1]{}
```

File 119 **lwarp-drftcite.sty**

§ 228 Package drftcite

(*Emulates or patches code by* Donald Arseneau.)

drftcite (*Pkg*) drftcite is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{drftcite}[1995/01/23]

\def\@lbibitem[#1]#2{\global\@HighCite\z@
  \item[
    \textsuperscript{\@nameuse{DCN@#2\@extra@b@citeb}}~%     lwarp
    \@biblabel{\@ifundefined{DCN@#2\@extra@b@citeb}{\@warning
    {Reference `#2' on page \thepage\space was never cited}}{}%
%    \DC@llap{$^{\@nameuse{DCN@#2\@extra@b@citeb}}$\ \ }%%o
   \@citeverb{#2}}\hfil]\if@filesw{\def\protect##1{\string ##1\space}%
   \immediate\write\@auxout{\string\bibcite{#2}{#1}}}\fi\ignorespaces}
```

File 120 **lwarp-easy-todo.sty**

§ 229 Package easy-todo

(*Emulates or patches code by* Juan Rada-Vilela.)

easy-todo (*Pkg*) easy-todo is patched for use by lwarp.

To remove the "P." heading for HTML:

```
\warpHTMLonly{\renewcommand{\todoindexpagetitle}{}}
```

for HTML output:

```
\LWR@ProvidesPackagePass{easy-todo}[2014/01/01]
```

`\listoftodos` Modified to correct buggy use of `\flushright`.

```
\let\LWR@easytodo@origlistoftodos\listoftodos

\renewcommand{\listoftodos}{%
\begingroup
\renewcommand{\flushright}{}
\LWR@easytodo@origlistoftodos
\endgroup
}
```

`\todoii` Modified to use `\textcolor` instead of `\color`.

```
\renewcommand{\todoii}[2]{%
\ifthenelse{\equal{\@todoobeyfinal}{true}}%
    {%
        \ifoptionfinal{\todoenable{false}}{\todoenable{true}}%
    }%
    {}%
\ifthenelse{\equal{\@todoenable}{true}}%
    {%
        \refstepcounter{todos}%
        \noindent{%
            \todocolor%
            \LWR@textcurrentcolor{%
                \normalfont\scriptsize{\bfseries{\thetodos.#1}}%
            }%
        }%
       \addcontentsline{lod}{todos}{\protect{\thetodos. }\LWR@isolate{#2}}%
    }%
    {}%
}
```

File 121 **`lwarp-ebook.sty`**

§ 230 Package ebook

(*Emulates or patches code by* Jørgen Steensgaard.)

ebook (*Pkg*) ebook is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{ebook}

\setcounter{secnumdepth}{0}
\setcounter{tocdepth}{2}

\providecommand{\pagefill}[1][0.001mm]{\noindent}

\providecommand{\ebook}{
\setcounter{secnumdepth}{0}
\setcounter{tocdepth}{2}
}
```

File 122 **lwarp-econometrics.sty**

§ 231 Package

econometrics

(*Emulates or patches code by* Erik Kole.)

`econometrics` (*Pkg*) econometrics is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{econometrics}% no date specified in the original

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{econometrics}

\CustomizeMathJax{\newcommand{\SC}{\mathbb{C}}}
\CustomizeMathJax{\newcommand{\SN}{\mathbb{N}}}
\CustomizeMathJax{\newcommand{\SQ}{\mathbb{Q}}}
\CustomizeMathJax{\newcommand{\SR}{\mathbb{R}}}
\CustomizeMathJax{\newcommand{\SZ}{\mathbb{Z}}}

\CustomizeMathJax{\newcommand{\calA}{\mathcal{A}}}
\CustomizeMathJax{\newcommand{\calB}{\mathcal{B}}}
\CustomizeMathJax{\newcommand{\calC}{\mathcal{C}}}
\CustomizeMathJax{\newcommand{\calD}{\mathcal{D}}}
\CustomizeMathJax{\newcommand{\calE}{\mathcal{E}}}
\CustomizeMathJax{\newcommand{\calF}{\mathcal{F}}}
\CustomizeMathJax{\newcommand{\calG}{\mathcal{G}}}
\CustomizeMathJax{\newcommand{\calH}{\mathcal{H}}}
\CustomizeMathJax{\newcommand{\calI}{\mathcal{I}}}
\CustomizeMathJax{\newcommand{\calJ}{\mathcal{J}}}
\CustomizeMathJax{\newcommand{\calK}{\mathcal{K}}}
\CustomizeMathJax{\newcommand{\calL}{\mathcal{L}}}
\CustomizeMathJax{\newcommand{\calM}{\mathcal{M}}}
\CustomizeMathJax{\newcommand{\calN}{\mathcal{N}}}
\CustomizeMathJax{\newcommand{\calO}{\mathcal{O}}}
\CustomizeMathJax{\newcommand{\calP}{\mathcal{P}}}
\CustomizeMathJax{\newcommand{\calQ}{\mathcal{Q}}}
\CustomizeMathJax{\newcommand{\calR}{\mathcal{R}}}
\CustomizeMathJax{\newcommand{\calS}{\mathcal{S}}}
\CustomizeMathJax{\newcommand{\calT}{\mathcal{T}}}
\CustomizeMathJax{\newcommand{\calU}{\mathcal{U}}}
\CustomizeMathJax{\newcommand{\calV}{\mathcal{V}}}
\CustomizeMathJax{\newcommand{\calW}{\mathcal{W}}}
\CustomizeMathJax{\newcommand{\calX}{\mathcal{X}}}
\CustomizeMathJax{\newcommand{\calY}{\mathcal{Y}}}
\CustomizeMathJax{\newcommand{\calZ}{\mathcal{Z}}}

\LWR@mathjax@addlatin@u@bfit{m}% uppercase Latin, bold italic
\LWR@mathjax@addlatin@l@bfit{v}% lowercase Latin, bold italic

\LWR@mathjax@addgreek@l@bfit{v}{}% lowercase Greek bold italic
\LWR@mathjax@addgreek@u@bfit*{m}{}% uppercase Greek bold italic, capitalized macro names

\CustomizeMathJax{\newcommand{\rb}{\mathrm{b}}}
\CustomizeMathJax{\newcommand{\rB}{\mathrm{B}}}
```

```
\CustomizeMathJax{\newcommand{\rC}{\mathrm{C}}}
\CustomizeMathJax{\newcommand{\rD}{\mathrm{D}}}
\CustomizeMathJax{\newcommand{\rf}{\mathrm{f}}}
\CustomizeMathJax{\newcommand{\rF}{\mathrm{F}}}
\CustomizeMathJax{\newcommand{\rH}{\mathrm{H}}}
\CustomizeMathJax{\newcommand{\rL}{\mathrm{L}}}
\CustomizeMathJax{\newcommand{\rN}{\mathrm{N}}}
\CustomizeMathJax{\newcommand{\rt}{\mathrm{t}}}
\CustomizeMathJax{\newcommand{\rU}{\mathrm{U}}}
\CustomizeMathJax{\newcommand{\rGam}{\mathrm{Gam}}}
\CustomizeMathJax{\newcommand{\rBeta}{\mathrm{Beta}}}

\CustomizeMathJax{\newcommand{\Bin}{\mathrm{Bin}}}
\CustomizeMathJax{\newcommand{\eu}{\mathrm{e}}}
\CustomizeMathJax{\newcommand{\iu}{\mathrm{i}}}
\CustomizeMathJax{\newcommand{\LN}{\mathrm{LN}}}
\CustomizeMathJax{\newcommand{\IN}{\mathrm{IN}}}

\CustomizeMathJax{\newcommand{\Poi}{\mathrm{Poi}}}

\CustomizeMathJax{\newcommand{\ped}[1]{_\mathrm{#1}}}
\CustomizeMathJax{\newcommand{\ap}[1]{^\mathrm{#1}}}
\CustomizeMathJax{\renewcommand{\Re}{\mathrm{Re}}{\nolimits}}
\CustomizeMathJax{\renewcommand{\Im}{\mathrm{Im}}{\nolimits}}

\CustomizeMathJax{\newcommand{\deriv}[3][]{%
    \frac{\mathrm{d}^{#1}#2}{\mathrm{d}\,#3^{#1}}%
}}
\CustomizeMathJax{\newcommand{\pderiv}[3][]{%
    \frac{\partial^{#1}#2}{\partial #3^{#1}}%
}}

\CustomizeMathJax{\newcommand{\bias}{\operatorname{bias}}}
\CustomizeMathJax{\newcommand{\col}{\operatorname{col}}}
\CustomizeMathJax{\newcommand{\corr}{\operatorname{corr}}}
\CustomizeMathJax{\newcommand{\cov}{\operatorname{cov}}}
\CustomizeMathJax{\newcommand{\dg}{\operatorname{dg}}}
\CustomizeMathJax{\newcommand{\diag}{\operatorname{diag}}}
\CustomizeMathJax{\newcommand{\E}{\operatorname{E}}}
\CustomizeMathJax{\newcommand{\etr}{\operatorname{etr}}}
\CustomizeMathJax{\newcommand{\ip}{\mathrm{int}}{\nolimits}}
\CustomizeMathJax{\newcommand{\kur}{\operatorname{kur}}}
\CustomizeMathJax{\newcommand{\MSE}{\operatorname{MSE}}}
\CustomizeMathJax{\newcommand{\MSFE}{\operatorname{MSFE}}}
\CustomizeMathJax{\newcommand{\OLS}{\operatorname{OLS}}}
\CustomizeMathJax{\newcommand{\plim}{\operatorname{plim}}}
\CustomizeMathJax{\newcommand{\resid}{\operatorname{resid}}}
\CustomizeMathJax{\newcommand{\rk}{\operatorname{rk}}}
\CustomizeMathJax{\newcommand{\SE}{\operatorname{SE}}}
\CustomizeMathJax{\newcommand{\sgn}{\operatorname{sgn}}}
\CustomizeMathJax{\newcommand{\tr}{\operatorname{tr}}}
\CustomizeMathJax{\newcommand{\var}{\operatorname{var}}}
\CustomizeMathJax{\renewcommand{\vec}{\operatorname{vec}}}
\CustomizeMathJax{\newcommand{\vech}{\operatorname{vech}}}

\CustomizeMathJax{\newcommand{\distr}{\sim}}
\CustomizeMathJax{\newcommand{\adistr}{\stackrel{a}{\distr}}}
\CustomizeMathJax{\newcommand{\diff}{\Delta}}
\CustomizeMathJax{\newcommand{\fdiff}{\diff_{\rf}}}
\CustomizeMathJax{\newcommand{\bdiff}{\diff_{\rb}}}
```

```
\CustomizeMathJax{\newcommand{\eps}{\epsilon}}
\CustomizeMathJax{\newcommand{\epsi}{\varepsilon}}

\CustomizeMathJax{\newcommand{\longto}{\longrightarrow}}
\CustomizeMathJax{\newcommand{\pto}{\stackrel{p}{\longrightarrow}}}
\CustomizeMathJax{\newcommand{\dto}{\stackrel{d}{\longrightarrow}}}
\CustomizeMathJax{\newcommand{\wto}{\stackrel{w}{\longrightarrow}}}

\CustomizeMathJax{\newcommand{\Infmat}{\bm\calI}}
\CustomizeMathJax{\newcommand{\Hesmat}{\bm\calH}}
\CustomizeMathJax{\newcommand{\bcdot}{\bullet}}

\CustomizeMathJax{\newcommand{\vones}{\bm\imath}}
\CustomizeMathJax{\newcommand{\vzeros}{\boldsymbol{0}}}
\CustomizeMathJax{\newcommand{\mZeros}{\mathbf{O}}}

\CustomizeMathJax{\newcommand{\e}{\eu}}
\CustomizeMathJax{\newcommand{\mply}{\cdot}}
\CustomizeMathJax{\newcommand{\rW}{\ensuremath{\mathrm{W}}}}
\end{warpMathJax}
```

File 123 **lwarp-ed.sty**

§ 232 Package

ed

(*Emulates or patches code by* Michael Kohlhase.)

ed (*Pkg*) ed is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{ed}[2012/01/29]
```

Bugs:

1. `todolist` fails with the `hide` option, as does `\edexplanation`.
2. `\edstubURI` is actually `\edstuURI`.

```
\RequirePackage{xcolor}

\renewenvironment{edstub}[2][The following blue text]
{%
    \def\@test{#1}%
    \begin{center}%
        \huge%
        \textcolor{red}{%
            #1 is only a provisional stub\\\Large
            the Office document
            \ifx\ed@stubURI\@empty{#2}\else\LWR@href{\ed@stubURI}{#2}\fi\
            contains more text\\which will be merged for the final document%
        }%
    \end{center}%
    \BlockClass[color:blue]{edstub}%
}
{\endBlockClass}
```

File 124 **lwarp-ellipsis.sty**

§ 233 Package ellipsis

(*Emulates or patches code by* Peter J. Heslin.)

ellipsis (*Pkg*) ellipsis is emulated.

```
\LWR@ProvidesPackageDrop{ellipsis}[2004/09/28]

\newcommand{\ellipsisgap}{0.1em}

\newcommand*{\midwordellipsis}{\,\textellipsis\,}
```

File 125 **lwarp-embrac.sty**

§ 234 Package embrac

(*Emulates or patches code by* Clemens Niederberger.)

embrac (*Pkg*) embrac is patched for HTML and used as-is for print.

for HTML output:

```
\LWR@ProvidesPackagePass{embrac}[2017/07/04]

\ExplSyntaxOn
\RenewDocumentCommand{\embrac_kern:n}{m}{}
\ExplSyntaxOff

\LetLtxMacro\LWR@orig@HTML@emph\LWR@HTML@emph
\RenewDocumentCommand{\LWR@HTML@emph}{s m}{\LWR@orig@HTML@emph{#2}}

\LetLtxMacro\LWR@orig@HTML@textit\LWR@HTML@textit
\RenewDocumentCommand{\LWR@HTML@textit}{s m}{\LWR@orig@HTML@textit{#2}}

\LetLtxMacro\LWR@orig@HTML@textsl\LWR@HTML@textsl
\RenewDocumentCommand{\LWR@HTML@textsl}{s m}{\LWR@orig@HTML@textsl{#2}}

\ifxetexorluatex
    \LetLtxMacro\LWR@orig@HTML@textsi\LWR@HTML@textsi
    \RenewDocumentCommand{\LWR@HTML@textsi}{s m}{%
    \LWR@orig@HTML@textsi{#2}}
\fi

\AtBeginDocument{
    \LWR@formatted{emph}
    \LWR@formatted{textit}
    \LWR@formatted{textsl}
    \ifxetexorluatex
        \LWR@formatted{textsi}
    \fi
}

```

```
\newcommand{\LWR@HTML@EmbracOff}{}
\LWR@formatted{EmbracOff}

\newcommand{\LWR@HTML@EmbracOn}{}
\LWR@formatted{EmbracOn}
```

File 126 **lwarp-emptypage.sty**

§ 235 Package emptypage

emptypage (*Pkg*) emptypage is ignored.

for HTML output: Discard all options for lwarp-emptypage:

```
\LWR@ProvidesPackageDrop{emptypage}[2010/05/30]
```

File 127 **lwarp-endfloat.sty**

§ 236 Package endfloat

endfloat (*Pkg*) endfloat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{endfloat}[2019/04/15]

\newcommand\figureplace{}
\newcommand\tableplace{}
\newcommand\floatplace[1]{}
\newcounter{posttable}
\newcounter{postfigure}
\newcommand*{\theposttbl}{}
\newcommand*{\thepostfig}{}
\newcommand{\AtBeginFigures}[1]{}
\newcommand{\AtBeginTables}[1]{}
\newcommand{\AtBeginDelayedFloats}[1]{}
\newcommand*{\processdelayedfloats}{}
\newcommand*{\efloatseparator}{}
\def\efloattype{}
\providecommand\efloatheading[1]{}
\providecommand\efloatpreamble{}
\providecommand\efloatpostamble{}
\NewDocumentCommand{\addtodelayedfloat}{s m m}{}
\providecommand{\efloatbegin}{}
\providecommand{\efloatend}{}
\providecommand{\efloatbeginlist}{}
\providecommand{\efloatendlist}{}
```

File 128 **lwarp-endheads.sty**

§ 237 Package endheads

endheads (*Pkg*) endheads is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{endheads}[2017/04/06]
```

```
\newcommand{\changesinglepageabbrev}[1]{}
\newcommand{\changemultiplepageabbrev}[1]{}
\newcommand{\changenotesname}[1]{}
\newcommand{\changenotesheader}[1]{}
\newcommand{\changenotescontentsname}[1]{}
\newcommand{\changechapternotesline}[1]{}
\newcommand{\checknoteheaders}{}
\newif\ifnotesincontentson \notesincontentsonfalse
\newcommand{\notesincontents}{\notesincontentsontrue}
\newif\ifendnoteheaderson \endnoteheadersonfalse
\newcommand{\setupendnoteheaders}{%
    \endnoteheadersontrue%
}
\newif\iftitleinnotes \titleinnotestrue
\newcommand{\styleforchapternotebegin}{}
\newcommand{\styleforchapternoteend}{}
\newcommand{\setstyleforchapternotebegin}[1]{%
    \renewcommand{\styleforchapternotebegin}{#1}%
}
\newcommand{\setstyleforchapternoteend}[1]{%
    \renewcommand{\styleforchapternoteend}{#1}%
}
\newcommand{\resetendnotes}{}
\newif\ifnotesbychapteron \notesbychapteronfalse
\newcommand{\notesbychapter}{\notesbychapterontrue}
```

File 129 **lwarp-endnotes.sty**

§ 238 Package

endnotes

(*Emulates or patches code by* John Lavagnino.)

endnotes (*Pkg*) Patched for HTML.

table of contents To place the endnotes in the TOC, use:

```
\usepackage{endnotes}
\appto\enoteheading{\addcontentsline{toc}{section}{\notesname}}
\renewcommand*{\notesname}{Endnotes} % optional
```

HTML page To additionally have the endnotes on their own HTML page, if `FileDepth` allows:

```
\ForceHTMLPage
\theendnotes
```

⚠ `\endnotemark` numbering If using MathJax, see section 8.5.4 regarding the use of `\endnotemark` and `\endnotetext`.

for HTML output:

```
\LWR@ProvidesPackagePass{endnotes}
```

```
\def\enoteformat{%
% \rightskip\z@ \leftskip\z@ \parindent=1.8em
\leavevmode
% \llap{
\makeenmark
% }
}
```

```
\def\LWR@HTML@@makeenmark{\hbox{\LWR@htmlspan{sup}{\normalfont\theenmark}}}
\LWR@formatted{@makeenmark}

\def\makeenmark{\@makeenmark}
```

To nullify the endnotes:

```
\apptocmd{\LWR@nullifyfootnotes}{%
    \renewcommand{\endnote}[2][]{}%
    \renewcommand{\endnotemark}[1]{}%
}{}{}
```

For MathJax:

```
\begin{warpMathJax}
\def\endnotename{endnote}
\appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRendnote}{\theendnote}}
\appto\LWR@syncnotenames{\LWR@synconenotename{LWRendnote}{\endnotename}}
\CustomizeMathJax{\def\LWRendnote{1}}
\CustomizeMathJax{\newcommand{\endnote}[2][\LWRendnote]{{}^{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\endnotemark}[1][\LWRendnote]{{}^{\mathrm{#1}}}}
\end{warpMathJax}
```

File 130 **lwarp-engtlc.sty**

§ 239 Package

engtlc

(*Emulates or patches code by* Claudio Fiandrino.)

engtlc (*Pkg*) engtlc is patched for use by lwarp. MathJax is emulated.

⚠ For MathJax, `\signt`, `\signf`, `\signn`, and `\signz` do not force letter case as they do in SVG math.

for HTML output:

```
\LWR@ProvidesPackagePass{engtlc}[2012/12/18]

\newcommand{\LWR@HTML@finees}{%
    \begin{BlockClass}[text-align:right]{exerend}%
    \HTMLunicode{220E}%
    \end{BlockClass}%
}
\LWR@formatted{finees}

\newcommand{\LWR@HTML@exerend}{\finees}
\LWR@formatted{exerend}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{engtlc}

\CustomizeMathJax{\newcommand{\unit}[1]{\,\mathrm{#1}}}
\CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
%
\CustomizeMathJax{\newcommand{\ho}{\unit{h}}}
\CustomizeMathJax{\newcommand{\s}{\unit{s}}}
\CustomizeMathJax{\newcommand{\ms}{\unit{ms}}}
\CustomizeMathJax{\newcommand{\us}{\unit{\micro s}}}
\CustomizeMathJax{\newcommand{\ns}{\unit{ns}}}
```

```
\CustomizeMathJax{\newcommand{\ps}{\unit{ps}}}
%
\CustomizeMathJax{\newcommand{\um}{\unit{\micro m}}}
\CustomizeMathJax{\newcommand{\mm}{\unit{mm}}}
\CustomizeMathJax{\newcommand{\cm}{\unit{cm}}}
\CustomizeMathJax{\newcommand{\dm}{\unit{dm}}}
\CustomizeMathJax{\newcommand{\m}{\unit{m}}}
\CustomizeMathJax{\newcommand{\km}{\unit{km}}}
%
\CustomizeMathJax{\newcommand{\MA}{\unit{MA}}}
\CustomizeMathJax{\newcommand{\kA}{\unit{kA}}}
\CustomizeMathJax{\newcommand{\A}{\unit{A}}}
\CustomizeMathJax{\newcommand{\mA}{\unit{mA}}}
\CustomizeMathJax{\newcommand{\uA}{\unit{\micro A}}}
\CustomizeMathJax{\newcommand{\nA}{\unit{nA}}}
%
\CustomizeMathJax{\newcommand{\MV}{\unit{MV}}}
\CustomizeMathJax{\newcommand{\kV}{\unit{kV }}}
\CustomizeMathJax{\newcommand{\V}{\unit{V}}}
\CustomizeMathJax{\newcommand{\mV}{\unit{mV}}}
\CustomizeMathJax{\newcommand{\uV}{\unit{\micro V}}}
%
\CustomizeMathJax{\newcommand{\mohm}{\unit{m\Omega}}}
\CustomizeMathJax{\newcommand{\ohm}{\unit{\Omega}}}
\CustomizeMathJax{\newcommand{\kohm}{\unit{k\Omega}}}
\CustomizeMathJax{\newcommand{\Mohm}{\unit{M\Omega}}}
%
\CustomizeMathJax{\newcommand{\pSi}{\unit{pS}}}
\CustomizeMathJax{\newcommand{\nSi}{\unit{nS}}}
\CustomizeMathJax{\newcommand{\uSi}{\unit{\micro S}}}
\CustomizeMathJax{\newcommand{\mSi}{\unit{mS}}}
\CustomizeMathJax{\newcommand{\Si}{\unit{S}}}
\CustomizeMathJax{\newcommand{\kSi}{\unit{kS}}}
\CustomizeMathJax{\newcommand{\MSi}{\unit{MS}}}
%
\CustomizeMathJax{\newcommand{\fFa}{\unit{fF}}}
\CustomizeMathJax{\newcommand{\pFa}{\unit{pF}}}
\CustomizeMathJax{\newcommand{\nFa}{\unit{nF}}}
\CustomizeMathJax{\newcommand{\uFa}{\unit{\micro F}}}
\CustomizeMathJax{\newcommand{\mFa}{\unit{mF}}}
\CustomizeMathJax{\newcommand{\Fa}{\unit{F}}}
%
\CustomizeMathJax{\newcommand{\fHe}{\unit{fH}}}
\CustomizeMathJax{\newcommand{\pHe}{\unit{pH}}}
\CustomizeMathJax{\newcommand{\nHe}{\unit{nH}}}
\CustomizeMathJax{\newcommand{\uHe}{\unit{\micro H}}}
\CustomizeMathJax{\newcommand{\mHe}{\unit{mH}}}
\CustomizeMathJax{\newcommand{\He}{\unit{H}}}
%
\CustomizeMathJax{\newcommand{\dB}{\unit{dB}}}
\CustomizeMathJax{\newcommand{\dBm}{\unit{dBm}}}
%
\CustomizeMathJax{\newcommand{\uW}{\unit{\micro W}}}
\CustomizeMathJax{\newcommand{\mW}{\unit{mW}}}
\CustomizeMathJax{\newcommand{\W}{\unit{W}}}
\CustomizeMathJax{\newcommand{\kW}{\unit{kW}}}
\CustomizeMathJax{\newcommand{\MW}{\unit{MW}}}
%
\CustomizeMathJax{\newcommand{\Hz}{\unit{Hz}}}
\CustomizeMathJax{\newcommand{\kHz}{\unit{kHz}}}
```

```
\CustomizeMathJax{\newcommand{\MHz}{\unit{MHz}}}
\CustomizeMathJax{\newcommand{\GHz}{\unit{GHz}}}
\CustomizeMathJax{\newcommand{\THz}{\unit{THz}}}
%
\CustomizeMathJax{\newcommand{\bit}{\unit{bit}}}
\CustomizeMathJax{\newcommand{\kbit}{\unit{Kib}}}
\CustomizeMathJax{\newcommand{\Mbit}{\unit{Mib}}}
\CustomizeMathJax{\newcommand{\Byte}{\unit{B}}}
\CustomizeMathJax{\newcommand{\kByte}{\unit{KiB}}}
\CustomizeMathJax{\newcommand{\MByte}{\unit{Mib}}}
\CustomizeMathJax{\newcommand{\GByte}{\unit{GiB}}}
\CustomizeMathJax{\newcommand{\TByte}{\unit{TiB}}}
\CustomizeMathJax{\newcommand{\bits}{\unit{bit/s}}}
\CustomizeMathJax{\newcommand{\kbits}{\unit{Kib/s}}}
\CustomizeMathJax{\newcommand{\Mbits}{\unit{Mib/s}}}
\CustomizeMathJax{\newcommand{\Bytes}{\unit{B/s}}}
\CustomizeMathJax{\newcommand{\kBytes}{\unit{KiB/s}}}
\CustomizeMathJax{\newcommand{\MBytes}{\unit{MiB/s}}}
\CustomizeMathJax{\newcommand{\GBytes}{\unit{GiB/s}}}
\CustomizeMathJax{\newcommand{\TBytes}{\unit{TiB/s}}}
\CustomizeMathJax{\newcommand{\chips}{\unit{chip/s}}}
\CustomizeMathJax{\newcommand{\kchips}{\unit{Ki\mkern2mu chip/s}}}
\CustomizeMathJax{\newcommand{\Mchips}{\unit{Mi\mkern2mu chip/s}}}
\CustomizeMathJax{\newcommand{\chipsubit}{\unit{chip/bit}}}
%
\CustomizeMathJax{\newcommand{\frecciadex}[1][0.5]{%
    \hspace{.25cm}\Longrightarrow \hspace{.25cm}}%
}
\CustomizeMathJax{\newcommand{\varianzarumore}{\frac{N_0}{2}}}
%
\CustomizeMathJax{\newcommand{\etsymbolbracearg}[2]{%
    #1\mathopen{}\left\lbrace#2\right\rbrace\mathclose{}}%
}
\CustomizeMathJax{\newcommand{\fourier}[1]{\etsymbolbracearg{\mathcal{F}}{#1}}}
\CustomizeMathJax{\newcommand{\invfourier}[1]{\etsymbolbracearg{\mathcal{F}^{-1}}{#1}}}
\CustomizeMathJax{\newcommand{\partereale}[1]{\etsymbolbracearg{\textbf{Re}}{#1}}}
\CustomizeMathJax{\newcommand{\parteimm}[1]{\etsymbolbracearg{\textbf{Im}}{#1}}}
\CustomizeMathJax{\newcommand{\Info}[1]{I\left(#1\right)}}
\CustomizeMathJax{\newcommand{\versore}[1]{\hat{#1}}}
\CustomizeMathJax{\newcommand{\vettore}[1]{\overrightarrow{#1}}}
\CustomizeMathJax{\newcommand{\coseno}[1]{\cos\left(2\pi#1t\right)}}
\CustomizeMathJax{\newcommand{\seno}[1]{\sin\left(2\pi#1t\right)}}
\CustomizeMathJax{\newcommand{\energia}[1]{\mathcal{E}_{#1}}}
\CustomizeMathJax{\newcommand{\moduloexp}[2]{\left\vert#1\right\vert^{#2}}}
\CustomizeMathJax{\newcommand{\modulo}[1]{\left\vert#1\right\vert}}
\CustomizeMathJax{\newcommand{\indB}[1]{%
    \mathopen{}\left.#1\right\vert_{\mathrm{dB}}\mathclose{}}}%
\CustomizeMathJax{\newcommand{\for}[2]{\left. #1 \right\vert_{#2}}}
\CustomizeMathJax{\newcommand{\massimo}[1]{\etsymbolbracearg{\max}{#1}}}
\CustomizeMathJax{\newcommand{\minimo}[1]{\etsymbolbracearg{\min}{#1}}}
\CustomizeMathJax{\newcommand{\valc}{3\cdot 10^8}}
\CustomizeMathJax{\newcommand{\loga}[2]{\log_{#1}#2}}
\CustomizeMathJax{\newcommand{\analitic}[1]{\mathring{#1}}}
\CustomizeMathJax{\newcommand{\diff}{\mathop{}\mathopen{\mathrm{d}}}}
\CustomizeMathJax{\newcommand{\intinf}[1]{\int_{-\infty}^{+\infty}{#1}}}
\CustomizeMathJax{\newcommand{\deltain}[1]{\delta\left(#1\right)}}
\CustomizeMathJax{\newcommand{\iu}{\mathrm{j}}}
\CustomizeMathJax{\newcommand{\ex}[1]{\mathrm{e}^{#1}}}
%
\CustomizeMathJax{\newcommand{\gammatens}{{}^{\mathrm{V}}\Gamma}}
```

```
\CustomizeMathJax{\newcommand{\gammacorr}{{}^{\mathrm{I}}\Gamma}}
\CustomizeMathJax{\newcommand{\gammatensin}[1]{{}^{\mathrm{V}}\Gamma_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\gammacorrin}[1]{{}^{\mathrm{I}}\Gamma_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\gammain}[1]{\Gamma_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\gammak}{{}^{\mathrm{k}}\Gamma}}
%
\CustomizeMathJax{\newcommand{\lbvt}{\lambda_0}}
\CustomizeMathJax{\newcommand{\lbg}{\lambda_g}}
\CustomizeMathJax{\newcommand{\lbgvt}{\lambda_{g_0}}}
%
\CustomizeMathJax{\newcommand{\potin}[1]{P_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\potdisp}[1][]{P_{\mathrm{disp}}^{#1}}}
\CustomizeMathJax{\newcommand{\potDC}[1][]{P_{\mathrm{DC}}^{#1}}}
\CustomizeMathJax{\newcommand{\potCC}[1][]{P_{\mathrm{CC}}^{#1}}}
\CustomizeMathJax{\newcommand{\potirr}[1][]{P_{\mathrm{irr}}^{#1}}}
\CustomizeMathJax{\newcommand{\potdiss}[1][]{P_{\mathrm{diss}}^{#1}}}
\CustomizeMathJax{\newcommand{\potinc}[1][]{P_{\mathrm{inc}}^{#1}}}
%
\CustomizeMathJax{\newcommand{\z}[1]{Z_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\znorm}[1]{z_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\y}[1]{Y_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\ynorm}[1]{y_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\zinf}[1][]{Z_{\infty#1}}}
\CustomizeMathJax{\newcommand{\zinfn}[1]{\zinf[#1]}}
\CustomizeMathJax{\newcommand{\yinf}[1][]{Y_{\infty#1}}}
\CustomizeMathJax{\newcommand{\yinfn}[1]{\yinf[#1]}}
\CustomizeMathJax{\newcommand{\zvt}{Z_0}}
\CustomizeMathJax{\newcommand{\yvt}{Y_0}}
%
\CustomizeMathJax{\newcommand{\campoe}{\underline{\mathcal{E}}(\underline{r},t)}}
\CustomizeMathJax{\newcommand{\campoefas}{\underline{E}(\underline{r})}}
\CustomizeMathJax{\newcommand{\campoh}{\underline{\mathcal{H}}(\underline{r},t)}}
\CustomizeMathJax{\newcommand{\campohfas}{\underline{H}(\underline{r})}}
%
\CustomizeMathJax{\newcommand{\signt}[1]{{#1}(t)}}
\CustomizeMathJax{\newcommand{\signf}[1]{{#1}(f)}}
\CustomizeMathJax{\newcommand{\signn}[1]{{#1}(n)}}
\CustomizeMathJax{\newcommand{\signz}[1]{{#1}(z)}}
%
\CustomizeMathJax{\newcommand{\prob}[1]{\mathcal{P}\left(#1\right)}}
\CustomizeMathJax{\newcommand{\valatt}[1]{\mathbb{E}\left[#1\right]}}
\CustomizeMathJax{\newcommand{\var}[1]{\mathrm{Var}\left[#1\right]}}
\CustomizeMathJax{\newcommand{\comma}{\, , \,}}
\CustomizeMathJax{\newcommand{\dato}{\,|\,}}
%
\CustomizeMathJax{\let\bfRe\partereale}
\CustomizeMathJax{\let\bfIm\parteimm}
\CustomizeMathJax{\let\noisevar\varianzarumore}
% \CustomizeMathJax{\let\exerend\finees}
\CustomizeMathJax{\let\Spimplies\frecciadex}
\CustomizeMathJax{\let\Downimplies\frecciadown}
\CustomizeMathJax{\let\unitvec\versore}
\CustomizeMathJax{\let\vector\vettore}
\CustomizeMathJax{\let\cosine\coseno}
\CustomizeMathJax{\let\sine\seno}
\CustomizeMathJax{\let\energy\energia}
\CustomizeMathJax{\let\Abs\modulo}
\CustomizeMathJax{\let\AbsPow\moduloexp}
\CustomizeMathJax{\let\Max\massimo}
\CustomizeMathJax{\let\Min\minimo}
```

```
\CustomizeMathJax{\let\clight\valc}
\CustomizeMathJax{\let\Log\loga}
\CustomizeMathJax{\let\analytic\analitic}
\CustomizeMathJax{\let\infint\intinf}
\CustomizeMathJax{\let\deltaimp\deltain}
\CustomizeMathJax{\let\Vgamma\gammatens}
\CustomizeMathJax{\let\Cgamma\gammacorr}
\CustomizeMathJax{\let\Vgammain\gammatensin}
\CustomizeMathJax{\let\Cgammain\gammacorrin}
\CustomizeMathJax{\let\Kgamma\gammak}
\CustomizeMathJax{\let\powerin\potin}
\CustomizeMathJax{\let\availpow\potdisp}
\CustomizeMathJax{\let\irrpow\potirr}
\CustomizeMathJax{\let\disspow\potdiss}
\CustomizeMathJax{\let\incpow\potinc}
\CustomizeMathJax{\let\potalim\potCC}
\CustomizeMathJax{\let\potDC\potCC}
\CustomizeMathJax{\let\Efield\campoe}
\CustomizeMathJax{\let\Hfield\campoh}
\CustomizeMathJax{\let\phasorEfield\campoefas}
\CustomizeMathJax{\let\phasorHfiled\campohfas}
\CustomizeMathJax{\let\given\dato}
\CustomizeMathJax{\let\expval\valatt}
\CustomizeMathJax{\let\rmexp\ex}
\end{warpMathJax}
```

File 131 **lwarp-enotez.sty**

§ 240 Package enotez

(*Emulates or patches code by* Clemens Niederberger.)

enotez (*Pkg*) enotez is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{enotez}[2020/12/13]
```

Hyperref is emulated by lwarp, so it is forced on for enotez:

```
\ExplSyntaxOn
\AtBeginDocument{
    \bool_set_true:N \l__enotez_hyperref_bool
    \bool_set_true:N \l__enotez_hyperfootnotes_bool
}
```

Do not move or \hbox the \hypertarget:

```
% typeset the actual mark:
% #1: id
% #2: mark
\cs_gset_protected:Npn \enotez_write_mark:nn #1#2
  {
    \bool_if:NTF \l__enotez_hyperfootnotes_bool
      {
        \enotezwritemark { \hyperlink {enz.#1} { \enmarkstyle #2 } }
        \bool_if:NT \l__enotez_hyperbackref_bool
          {
%             \box_move_up:nn {1em} {
```

```
%                 \hbox:n {
                    \hypertarget {enz.#1.backref} { }
%                 }
%             }
          }
      }
      { \enotezwritemark { \enmarkstyle #2 } }
  }
\cs_generate_variant:Nn \enotez_write_mark:nn {x}
```

Do not move or \hbox the \hypertarget:

```
\cs_gset_protected:Npn \enotez_write_list_number:n #1
  {
    \bool_if:NT \l__enotez_hyperfootnotes_bool
      {
%         \box_move_up:nn {1em} { \hbox:n {
            \hypertarget {enz.#1} { }
%         } }
      }
    \tl_use:N \l__enotez_list_number_format_tl
    \tl_if_eq:nxTF {a} { \prop_item:Nn \g__enotez_endnote_man_prop {#1} }
      {
        \bool_if:nTF
          { \l__enotez_hyperfootnotes_bool && \l__enotez_hyperbackref_bool }
          {
            \exp_args:Nnx
            \hyperlink {enz.#1.backref}
              { \exp_not:V \l__enotez_endnote_mark_tl }
          }
          { \prop_item:Nn \g__enotez_endnote_mark_prop {#1} }
      }
      {
        \bool_if:nTF
          { \l__enotez_hyperfootnotes_bool && \l__enotez_hyperbackref_bool }
          {
            \exp_args:Nnx
            \hyperlink {enz.#1.backref}
              { \exp_not:V \l__enotez_endnote_mark_tl }
          }
          { \tl_use:N \l__enotez_endnote_mark_tl }
      }
  }
```

Do not move the label to the left:

```
\DeclareTemplateCode {enotez-list} {paragraph} {1}
  {
    heading       = \enotez_list_heading:n           ,
    format        = \l__enotez_list_format_tl        ,
    number        = \enotez_list_number:n            ,
    number-format = \l__enotez_list_number_format_tl ,
    notes-sep     = \l__enotez_list_notes_sep_dim
  }
  {
    \AssignTemplateKeys
    \enotez_set_totoc:
    \enotez_list_heading:n { \l__enotez_list_name_tl }
    \enotez_list_preamble:
    \enotez_build_print_list:nnnn {#1}
```

```
      {}
      {
        \par\noindent
        \group_begin:
          \tl_use:N \l__enotez_list_format_tl
%           \hbox_overlap_left:n
%             {
            \enotez_list_number:n
              { \enotez_write_list_number:n {##1} }
            \tl_use:N \c_space_tl
%             }
          % \cs_set:cpn {@currentlabel}
          %   { \p@endnote \l__enotez_endnote_mark_tl }
          \tl_use:N \g__enotez_endnote_text_tl
          \par
          \dim_compare:nT { \l__enotez_list_notes_sep_dim != 0pt }
            { \addvspace { \l__enotez_list_notes_sep_dim } }
        \group_end:
      }
      {}
    \enotez_list_postamble:
  }

\ExplSyntaxOff
```

For MathJax:

```
\begin{warpMathJax}
\def\endnotename{endnote}
\appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRendnote}{\theendnote}}
\appto\LWR@syncnotenames{\LWR@synconenotename{LWRendnote}{\endnotename}}
\CustomizeMathJax{\def\LWRendnote{1}}
\CustomizeMathJax{\newcommand{\endnote}[2][\LWRendnote]{{}^{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\endnotemark}[1][\LWRendnote]{{}^{\mathrm{#1}}}}
\end{warpMathJax}
```

File 132 **lwarp-enumerate.sty**

§ 241 Package enumerate

`enumerate` (*Pkg*) enumerate is supported with no changes.

This package is only required because it was used in the past to drop and then emulate the package. It cannot be removed because an older version which dropped the package may still remain, for example in a local vs. distribution directory, but it is now supported directly by lwarp and thus must no longer be dropped.

for HTML output:

```
\LWR@ProvidesPackagePass{enumerate}[2015/07/23]
```

File 133 **lwarp-enumitem.sty**

§ 242 Package enumitem

(*Emulates or patches code by* Javier Bezos.)

enumitem (*Pkg*) enumitem is supported with minor adjustments.

for HTML output:
```
\LWR@ProvidesPackagePass{enumitem}[2018/11/30]
```

\newlist {⟨*name*⟩} {⟨*type*⟩} {⟨*maxdepth*⟩}
\renewlist {⟨*name*⟩} {⟨*type*⟩} {⟨*maxdepth*⟩}

For enumitem lists, new lists must have the start and end actions assigned to the new environment. Renewed lists already have their actions assigned, and thus need no changes.

```
\let\LWR@enumitem@orignewlist\newlist

\renewcommand*{\newlist}[3]{%
\LWR@enumitem@orignewlist{#1}{#2}{#3}%
\AtBeginEnvironment{#1}{\@nameuse{LWR@#2start}}%
\AtEndEnvironment{#1}{\@nameuse{LWR@#2end}}%
}

\def\DrawEnumitemLabel{}
```

File 134 **lwarp-epigraph.sty**

§ 243 Package epigraph

(*Emulates or patches code by* Peter Wilson.)

epigraph (*Pkg*) epigraph is emulated for HTML, and used as-is for print output.

Use CSS to format epigraphs.

for HTML output:
```
\LWR@ProvidesPackagePass{epigraph}[2020/01/02]


\DeclareDocumentCommand{\LWR@HTML@qitem}{m m}
{%
    \begin{BlockClass}{qitem}%
    #1%
    \LWR@stoppars%
    \ifbool{FormatWP}%
        {\begin{BlockClass}[border-top:1px solid gray]{epigraphsource}}%
        {\begin{BlockClass}{epigraphsource}}%
    #2%
    \end{BlockClass}%
    \end{BlockClass}%
}
\LWR@formatted{qitem}
```

epigraph: Added ARIA role.

```
\DeclareDocumentCommand{\LWR@HTML@epigraph}{m m}
{%
  \begin{LWR@BlockClassWP}{\LWR@print@mbox{text-align:right}}{}(note){epigraph}%
    \qitem{#1}{#2}%
    \end{LWR@BlockClassWP}%
}
\LWR@formatted{epigraph}
```

```
\DeclareDocumentEnvironment{LWR@HTML@epigraphs}{}
  {\LWR@BlockClassWP{\LWR@print@mbox{text-align:right}}{}(note){epigraph}}%
    {\endLWR@BlockClassWP}
\LWR@formattedenv{epigraphs}
```

The following cannot be used in print mode while generating HTML:

```
\renewcommand{\epigraphhead}[2][0]{#2}
\renewcommand{\dropchapter}[1]{}
\renewcommand*{\undodrop}{}
```

File 135 **lwarp-epsf.sty**

§ 244 Package epsf

(*Emulates or patches code by* Tom Rokicki.)

epsf (*Pkg*) epsf is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{epsf}% not date given

\xpretocmd{\epsfsetgraph}
    {\begin{lateximage}}
    {}
    {\LWR@patcherror{lwarp-epsf}{epsfsetgraph-begin}}

\xapptocmd{\epsfsetgraph}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{lwarp-epsf}{epsfsetgraph-end}}
```

File 136 **lwarp-epsfig.sty**

§ 245 Package epsfig

epsfig (*Pkg*) epsfig is emulated for use by lwarp.

⚠ Only the LaTeX2e syntax is emulated.

for HTML output:

```
\LWR@ProvidesPackagePass{epsfig}[2017/06/25]
```

A few additional keys to capture the filename:

```
\RequirePackage{graphics}

\define@key{igraph}{file}{%
    \xdef\LWR@epsfig@filename{#1}%
}

\define@key{igraph}{figure}{%
    \xdef\LWR@epsfig@filename{#1}%
}

```

```
\define@key{igraph}{prolog}{}

\define@key{igraph}{silent}[]{}
```

The captured filename is used as the argument to `\includegraphics`:

```
\newcommand{\LWR@HTML@epsfig}[1]{\includegraphics[#1]{\LWR@epsfig@filename}}
\LWR@formatted{epsfig}

\newcommand{\LWR@HTML@psfig}[1]{\includegraphics[#1]{\LWR@epsfig@filename}}
\LWR@formatted{psfig}
```

File 137 **lwarp-epstopdf.sty**

§ 246 Package epstopdf

epstopdf (*Pkg*)

Previous versions of lwarp had a nullfied version, but now epstopdf-base is supported. lwarp-epstopdf becomes a placeholder to overwrite previous versions.

See package epstopdf-base for details.

for HTML output:

```
\LWR@ProvidesPackagePass{epstopdf}[2020-01-24]
```

File 138 **lwarp-epstopdf-base.sty**

§ 247 Package epstopdf-base

epstopdf-base (*Pkg*)

⚠ convert to .svg

Images with an .eps extension will be converted to .pdf. The HTML output uses the .svg version, so use

Enter ⇒ **lwarpmk pdftosvg <listofPDFfiles>**

to generate .svg versions.

for HTML output:

```
\LWR@ProvidesPackagePass{epstopdf-base}[2020-01-24]
```

Redefine to remember the image filename, replacing .pdf with .svg. Use the epstopdf print version inside a `lateximage`.

```
\newcommand*{\LWR@HTML@ETE@OrgGin@setfile}[3]{%
    \edef\LWR@tempone{#3}%
    \StrSubstitute{\LWR@tempone}{.pdf}{.svg}[\LWR@tempone]%
    \StrSubstitute{\LWR@tempone}{.PDF}{.SVG}[\LWR@tempone]%
    \xdef\LWR@parsedfilename{\LWR@tempone}%
}

\LWR@formatted{ETE@OrgGin@setfile}
```

`\includegraphics` in HTML mode redefines `\Gin@setfile` to be `\LWR@HTML@Gin@setfile`, which is now redirected to epstopdf's version:

```
\renewcommand*{\LWR@HTML@Gin@setfile}[3]{%
    \ETE@Gin@setfile{#1}{#2}{#3}%
}
```

Allow .eps images to be found if a suffix is not provided:

```
\AtBeginDocument{
\DeclareGraphicsExtensions{%
    .eps,.EPS,.svg,.SVG,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
}
\DeclareGraphicsRule{.svg}{svg}{.svg}{}
\DeclareGraphicsRule{.SVG}{svg}{.SVG}{}
}
```

Likewise when inside a `lateximage`:

```
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{%
    .eps,.EPS,.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
}%
}
```

File 139 **lwarp-eqlist.sty**

§ 248 Package eqlist

`eqlist` (*Pkg*) eqlist is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{eqlist}[2002/08/15]
```

```
\newenvironment{eqlist}[1][]{\description}{\enddescription}
\newenvironment{eqlist*}[1][]{\description}{\enddescription}
\newenvironment{Eqlist}[2][]{\description}{\enddescription}
\newenvironment{Eqlist*}[2][]{\description}{\enddescription}
\newcommand*{\longitem}[1][]{\item[#1]}
\newcommand*{\eqlistinit}{}
\newcommand*{\eqliststarinit}{}
\newcommand*{\eqlistinitpar}{}
\def\eqlistlabel#1{#1}
\newcommand{\eqlistauto}[1]{}
\newcommand{\eqlistnoauto}{}
```

File 140 **lwarp-eqparbox.sty**

§ 249 Package eqparbox

(*Emulates or patches code by* Scott Pakin.)

`eqparbox` (*Pkg*) eqparbox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{eqparbox}[2017/09/03]
```

```
\NewDocumentCommand{\LWR@HTML@eqparbox}{O{t} O{} O{t} m +m}{%
```

```
    {%
        \minipagefullwidth%
        \parbox[#1][#2][#3]{\linewidth}{#5}%
    }%
}
\LWR@formatted{eqparbox}

\NewDocumentCommand{\LWR@HTML@eqmakebox}{o o m}{%
    \makebox[#2]{#3}%
}
\LWR@formatted{eqmakebox}

\NewDocumentCommand{\LWR@HTML@eqframebox}{o o m}{%
    \framebox[#2]{#3}%
}
\LWR@formatted{eqframebox}

\NewDocumentEnvironment{LWR@HTML@eqminipage}{O{t} O{} O{t} m}
{%
    \begingroup%
    \minipagefullwidth%
    \minipage[#1][#2][#3]{\linewidth}%
}%
{%
    \endminipage%
    \endgroup%
}

\newcommand*{\LWR@HTML@eqboxwidth}[1]{.25\linewidth}
\LWR@formatted{eqboxwidth}

\newcommand*{\LWR@HTML@eqsetminwidth}[2]{}
\newcommand*{\LWR@HTML@eqsetmaxwidth}[2]{}

\newcommand*{\LWR@HTML@eqsetminwidthto}[2]{}
\newcommand*{\LWR@HTML@eqsetmaxwidthto}[2]{}
```

File 141 **lwarp-errata.sty**

§ 250 Package errata

(*Emulates or patches code by* Michael Kohlhase.)

errata (*Pkg*) errata is patched for use by lwarp.

This is for v0.3 of errata. A newer version of errata with more features is under development, at which time the lwarp version will have to be updated.

for HTML output: Macros are being defined with the math dollar, so enable the HTML version during package loading:

```
\StartDefiningMath
```

Now load the package:

```
\LWR@ProvidesPackagePass{errata}[2006/11/12]
```

Patches for dynamic inline math:

```
\xpatchcmd{\erratumAdd}
    {$_a^{\arabic{erratum}}$}
%     {\inlinemathother$_a^{\arabic{erratum}}$\inlinemathnormal}
    {\textsubscript{a}\textsuperscript{\arabic{erratum}}}
    {}
    {\LWR@patcherror{erratum}{erratumAdd}}

\xpatchcmd{\erratumDelete}
    {$_d^{\arabic{erratum}}$}
%     {\inlinemathother$_d^{\arabic{erratum}}$\inlinemathnormal}
    {\textsubscript{d}\textsuperscript{\arabic{erratum}}}
    {}
    {\LWR@patcherror{erratum}{erratumDelete}}

\xpatchcmd{\erratumReplace}
    {$_r^{\arabic{erratum}}$}
%     {\inlinemathother$_r^{\arabic{erratum}}$\inlinemathnormal}
    {\textsubscript{r}\textsuperscript{\arabic{erratum}}}
    {}
    {\LWR@patcherror{erratum}{erratumReplace}}

\xpatchcmd{\erratum}
    {$_a$}
%     {\inlinemathother$_a$\inlinemathnormal}
    {\textsubscript{a}}
    {}
    {\LWR@patcherror{erratum}{erratumDelete}}

\xpatchcmd{\erratum}
    {$_d^{\@thefnmark}$}
%     {\inlinemathother$_d^{\@thefnmark}$\inlinemathnormal}
    {\textsubscript{d}\textsuperscript{\@thefnmark}}
    {}
    {\LWR@patcherror{erratum}{eDelete}}

\xpatchcmd{\erratum}
    {$_r^{\@thefnmark}$}
%     {\inlinemathother$_r^{\@thefnmark}$\inlinemathnormal}
    {\textsubscript{r}\textsuperscript{\@thefnmark}}
    {}
    {\LWR@patcherror{erratum}{eReplace}}
```

Finish the current page's errata before closing and reloading the list:

```
\preto\PrintErrata{\LWR@maybe@orignewpage}
```

No longer defining math macros with the HTML $:

```
\StopDefiningMath
```

File 142 **lwarp-eso-pic.sty**

§ 251 Package eso-pic

(*Emulates or patches code by* ROLF NIEPRASCHK.)

eso-pic (*Pkg*) eso-pic is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{eso-pic}[2018/04/12]

\newcommand*{\LenToUnit}{}
\newcommand{\AtPageUpperLeft}[1]{}
\newcommand{\AtPageLowerLeft}[1]{}
\newcommand{\AtPageCenter}[1]{}
\newcommand{\AtStockLowerLeft}[1]{}
\newcommand{\AtStockUpperLeft}[1]{}
\newcommand{\AtStockCenter}[1]{}
\newcommand{\AtTextUpperLeft}[1]{}
\newcommand{\AtTextLowerLeft}[1]{}
\newcommand{\AtTextCenter}[1]{}
\NewDocumentCommand{\AddToShipoutPictureBG}{s +m}{}

\newcommand{\AddToShipoutPicture}{\AddToShipoutPictureBG}
\NewDocumentCommand{\AddToShipoutPictureFG}{s +m}{}
\newcommand*{\ClearShipoutPictureBG}{}
\newcommand*{\ClearShipoutPicture}{}
\newcommand*{\ClearShipoutPictureFG}{}
\newcommand{\gridSetup}[6][]{}
```

File 143 **lwarp-esvect.sty**

§ 252 Package esvect

(*Emulates or patches code by* Eddie Saudrais.)

esvect (*Pkg*) esvect is used as-is for SVG math, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{esvect}% no date given

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\LWResvectvv}[1]{\overrightarrow{#1}}}
\CustomizeMathJax{\newcommand{\LWResvectvvstar}[2]{\overrightarrow{#1}\!_{#2}}}
\CustomizeMathJax{\newcommand{\vv}{\ifstar\LWResvectvvstar\LWResvectvv}}
\end{warpMathJax}
```

File 144 **lwarp-etoc.sty**

§ 253 Package etoc

etoc (*Pkg*) etoc is ignored. All commands are nullified.

⚠ `\tableofcontents` with `\ref`

The etoc package uses a non-standard syntax which looks ahead after a `\tableofcontents` for a following `\ref`. These `\ref`s appear in the HTML result unless they are removed. Where a `\tableofcontents` is followed by `\ref`, and perhaps also `\label` as well, enclose all of them inside `\warpprintonly`:

```
\warpprintonly{\tableofcontents                    \ref{toc:abc}
\label{toc:def}}
```

or place all code related to a local `\tableofcontents` inside a `warpprint` environment.

⚠ home page Be sure to keep the initial `\tableofcontents` on the home page, perhaps in its own `\warpHTMLonly` macro or `warpHTML` environment.

for HTML output:

```
\LWR@ProvidesPackageDrop{etoc}[2019/11/17]

\def\etocsetlevel#1#2{}
\def\etocskipfirstprefix{}
\let\etocthename   \@empty
\let\etocthenumber \@empty
\let\etocthepage   \@empty
\let\etocthelinkedname   \@empty
\let\etocthelinkednumber \@empty
\let\etocthelinkedpage   \@empty
\let\etocthelink   \@firstofone % prior to 1.08j its was \let to \@empty
\DeclareRobustCommand*{\etocname}  {}
\DeclareRobustCommand*{\etocnumber}{}
\DeclareRobustCommand*{\etocpage}  {}
\DeclareRobustCommand*{\etoclink}  {\@firstofone}
\DeclareRobustCommand*{\etocifnumbered}{\@firstoftwo}
\DeclareRobustCommand*{\etociffirst}{\@firstoftwo}
\DeclareRobustCommand*\etocifwasempty{\@firstoftwo}
\let\etocaftertitlehook    \@empty
\let\etocaftercontentshook \@empty
\def\etoctableofcontents{}
\newcommand*\localtableofcontents{}
\newcommand*\localtableofcontentswithrelativedepth[1]{}
\newcommand\etocsettocstyle[2]{}
\long\def\etocsetstyle#1#2#3#4#5{}
\def\etocfontminustwo {\normalfont \LARGE \bfseries}
\def\etocfontminusone {\normalfont \large \bfseries}
\def\etocfontzero     {\normalfont \large \bfseries}
\def\etocfontone      {\normalfont \normalsize \bfseries}
\def\etocfonttwo      {\normalfont \normalsize}
\def\etocfontthree    {\normalfont \footnotesize}
\def\etocsepminustwo  {4ex \@plus .5ex \@minus .5ex}
\def\etocsepminusone  {4ex \@plus .5ex \@minus .5ex}
\def\etocsepzero      {2.5ex \@plus .4ex \@minus .4ex}
\def\etocsepone       {1.5ex \@plus .3ex \@minus .3ex}
\def\etocseptwo       {.5ex \@plus .1ex \@minus .1ex}
\def\etocsepthree     {.25ex \@plus .05ex \@minus .05ex}
\def\etocbaselinespreadminustwo {1}
\def\etocbaselinespreadminusone {1}
\def\etocbaselinespreadzero     {1}
\def\etocbaselinespreadone      {1}
\def\etocbaselinespreadtwo      {1}
\def\etocbaselinespreadthree    {.9}
\def\etocminustwoleftmargin  {1.5em plus 0.5fil}
\def\etocminustworightmargin {1.5em plus -0.5fil}
\def\etocminusoneleftmargin  {1em}
\def\etocminusonerightmargin {1em}
\def\etoctoclineleaders
        {\hbox{\normalfont\normalsize\hb@xt@2ex {\hss.\hss}}}
\def\etocabbrevpagename {p.~}
\def\etocpartname       {Part}% modified 1.08b
\def\etocbookname       {Book}
\def\etocdefaultlines{}
\def\etocabovetocskip{3.5ex \@plus 1ex \@minus .2ex}
```

```
\def\etocbelowtocskip{3.5ex \@plus 1ex \@minus .2ex}
\def\etoccolumnsep{2em}
\def\etocmulticolsep{0ex}
\def\etocmulticolpretolerance{-1}
\def\etocmulticoltolerance{200}
\def\etocdefaultnbcol{2}
\def\etocinnertopsep{2ex}
\newcommand\etocmulticolstyle[2][]{}
\def\etocinnerbottomsep{3.5ex}
\def\etocinnerleftsep{2em}
\def\etocinnerrightsep{2em}
\def\etoctoprule{\hrule}
\def\etocleftrule{\vrule}
\def\etocrightrule{\vrule}
\def\etocbottomrule{\hrule}
\def\etoctoprulecolorcmd{\relax}
\def\etocbottomrulecolorcmd{\relax}
\def\etocleftrulecolorcmd{\relax}
\def\etocrightrulecolorcmd{\relax}
\newcommand*\etocruledstyle[2][]{}
\def\etocframedmphook{\relax}
\long\def\etocbkgcolorcmd{\relax}
\newcommand*\etocframedstyle[2][]{}
\def\etocmulticol{}
\def\etocruled{}
\def\etocframed{}
\def\etoclocalmulticol{}
\def\etoclocalruled{}
\def\etoclocalframed{}
\def\etocarticlestyle{}
\def\etocarticlestylenomarks{}
\def\etocbookstyle{}
\def\etocbookstylenomarks{}
\let\etocreportstyle\etocbookstyle
\let\etocreportstylenomarks\etocbookstylenomarks
\def\etocmemoirtoctotocfmt #1#2{}
\def\etocmemoirstyle{}
\def\etocscrartclstyle{}
\let\etocscrbookstyle\etocscrartclstyle
\let\etocscrreprtstyle\etocscrartclstyle
\def\etocstandarddisplaystyle{\etocarticlestyle}
\newcommand*\etocmarkboth[1]{}
\newcommand*\etocmarkbothnouc[1]{}
\newcommand\etoctocstyle[3][section]{}
\newcommand\etoctocstylewithmarks[4][section]{}
\newcommand\etoctocstylewithmarksnouc[4][section]{}
\def\etocignoretoctocdepth{}
\def\etocsettocdepth[1]{}
\def\etocdepthtag   #1#{\Etoc@depthtag }
\def\Etoc@depthtag  #1{}
\def\etocignoredepthtags {}
\def\etocobeydepthtags   {}
\def\etocsettagdepth #1#2{}
\def\invisibletableofcontents {}
\def\invisiblelocaltableofcontents{}
\def\etocsetnexttocdepth #1{}
\def\etocsetlocaltop #1#{\Etoc@set@localtop}
\def\Etoc@set@localtop #1{}
\def\etocstandardlines {}
\def\etoctoclines      {}
```

```
\let\etocaftertochook    \@empty
\let\etocbeforetitlehook \@empty
\appto\tableofcontents{\def\tableofcontents{}}
```

File 145 **lwarp-eurosym.sty**

§ 254 Package eurosym

(*Emulates or patches code by* Henrik Theiling.)

eurosym (*Pkg*) eurosym is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{eurosym}[1998/08/06]

\renewrobustcmd\officialeuro{\HTMLentity{euro}}
\let\geneuro\officialeuro
\let\geneuronarrow\officialeuro
\let\geneurowide\officialeuro
\let\euro\officialeuro
\renewrobustcmd\eurobars{}
\renewrobustcmd\eurobarsnarrow{}
\renewrobustcmd\eurobarswide{}
```

File 146 **lwarp-everypage.sty**

§ 255 Package everypage

(*Emulates or patches code by* Sergio Callegari.)

everypage (*Pkg*) everypage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{everypage}[2007/06/20]

\newcommand*{\AddEverypageHook}[1]{}
\newcommand*{\AddThispageHook}[1]{}
```

File 147 **lwarp-everyshi.sty**

§ 256 Package everyshi

(*Emulates or patches code by* Martin Schröder.)

everyshi (*Pkg*) ignored.

for HTML output: Discard all options for lwarp-everyshi:

```
\LWR@ProvidesPackageDrop{everyshi}[2001/05/15]

\let\EveryShipout\relax
\newcommand*{\EveryShipout}[1]{}

\let\AtNextShipout\relax
\newcommand*{\AtNextShipout}[1]{}
```

File 148 **lwarp-extarrows.sty**

§ 257 Package extarrows

(*Emulates or patches code by* Huynh Ky Anh.)

extarrows (*Pkg*) extarrows is used as-is for SVG math, and emulted for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{extarrows}[2008/05/15]

\begin{warpMathJax}
\CustomizeMathJax{\Newextarrow\xLongleftarrow{10,10}{0x21D0}}
\CustomizeMathJax{\Newextarrow\xLongrightarrow{10,10}{0x21D2}}
\CustomizeMathJax{\Newextarrow\xLongleftrightarrow{10,10}{0x21D4}}
\CustomizeMathJax{\Newextarrow\xLeftrightarrow{10,10}{0x21D4}}
\CustomizeMathJax{\Newextarrow\xlongleftrightarrow{10,10}{0x2194}}
\CustomizeMathJax{\Newextarrow\xleftrightarrow{10,10}{0x2194}}
\CustomizeMathJax{\let\xlongleftarrow\xleftarrow}
\CustomizeMathJax{\let\xlongrightarrow\xrightarrow}
\end{warpMathJax}
```

File 149 **lwarp-extramarks.sty**

§ 258 Package extramarks

(*Emulates or patches code by* Piet van Oostrum.)

extramarks (*Pkg*) extramarks is ignored.

for HTML output: Discard all options for lwarp-extramarks:

```
\LWR@ProvidesPackageDrop{extramarks}[2019/01/31]

\newcommand*{\extramarks}[2]{}
\newcommand*{\firstleftxmark}{}
\newcommand*{\lastleftxmark}{}
\newcommand*{\firstrightxmark}{}
\newcommand*{\lastrightxmark}{}
\newcommand*{\firstxmark}{}
\newcommand*{\lastxmark}{}
\newcommand*{\topxmark}{}
\newcommand*{\topleftxmark}{}
\newcommand*{\toprightxmark}{}
\newcommand*{\firstleftmark}{}
\newcommand*{\lastrightmark}{}
\newcommand*{\firstrightmark}{}
\newcommand*{\lastleftmark}{}
```

File 150 **lwarp-fancybox.sty**

§ 259 Package

fancybox

(*Emulates or patches code by* Timothy Van Zandt.)

fancybox (*Pkg*) fancybox is supported with some patches.

framed equation example fancybox's documentation has an example FramedEqn environment which combines math, \Sbox, a minipage, and an \fbox. This combination requires that the entire environment be enclosed inside a lateximage, which is done by adding \lateximage at the very start of FramedEqn's beginning code, and \endlateximage at the very end of the ending code. Unfortunately, the HTML alt attribute is not used here.

```
\newenvironmentFramedEqn
{
\lateximage% NEW
\setlength{\fboxsep}{15pt}
. . . }{. . .
\[\fbox{\TheSbox}\]
\endlateximage% NEW
}
```

framing alternatives \fbox works with fancybox. Also see lwarp's \fboxBlock macro and fminipage environment for alternatives to \fbox for framing environments.

framed table example The fancybox documentation's example of a framed table using an \fbox containing a tabular does not work with lwarp, but the FramedTable environment does work if \fbox is replaced by \fboxBlock. This method does lose some HTML formatting. A better method is to enclose the table's contents inside a fminipage environment. The caption may be placed either inside or outside the fminipage:

```
\begin{table}
\begin{fminipage}{\linewidth}
\begin{tabular}{lr}
. . .
\end{tabular}
\end{fminipage}
\end{table}
```

⚠ framed verbatim lwarp does not support the verbatim environment inside a span, box, or fancybox's \Sbox, but a verbatim may be placed inside a fminipage. The fancybox documentation's example FramedVerb may be defined as:

```
\newenvironment{FramedVerb}[1] % width
{
  \VerbatimEnvironment
  \fminipage{#1}
  \beginVerbatim
}{
  \endVerbatim
  \endfminipage
}
```

framed \VerbBox

fancybox's \VerbBox may be used inside \fbox.

indented alignment

LVerbatim, \LVerbatimInput, and \LUseVerbatim indent with horizontal space which may not line up exactly with what *pdftotext* detects. Some lines may be off slightly in their left edge.

fancybox, fancyvrb

⚠ \VerbatimFootnotes

⚠ sectioning or displaymath

If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

At present there is a bug such that paragraph closing tags are not present in footnotes when \VerbatimFootnotes are selected. The browser usually compensates.

```
\LWR@ProvidesPackagePass{fancybox}[2010/05/15]
```

After the preamble is loaded, after any patches to Verbatim:

```
\AfterEndPreamble{
\LWR@traceinfo{Patching fancybox.}
```

\VerbatimFootnotes

Patched to use the new version.

```
\def\VerbatimFootnotes{%
\let\@footnotetext\V@footnotetext%
\let\LWR@footnotetext\V@footnotetext% lwarp
}
```

\V@@footnotetext

Patches in a subset of lwarp's \LWR@footnotetext to the fancyvrb version of \V@@footnotetext.

```
\def\V@@footnotetext{%
\LWR@traceinfo{V@footnotetext}%
```

Place an autopage marker so that back references to citations inside a footnote will link closer to the footnote text, if possible.

```
\LWR@newautopagelabel{page}%
```

Take the current footnote box, then append:

```
\global\setbox\LWR@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
    \unvbox\LWR@footnotebox%
```

Remember the footnote number for \ref:

```
    \protected@edef\@currentlabel{%
        \csname p@footnote\endcsname\@thefnmark%
    }% @currentlabel
```

Use HTML superscripts in the footnote even inside a lateximage:

```
    \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
    \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Append the footnote to the list:

```
    \@makefntext{}%
```

The footnote text will follow after `\V@@@footnotetext` has completed.

```
  \bgroup%
  \aftergroup\V@@@footnotetext%
```

Do not generate autopages inside the footnotes, since they are accumulated at the moment before finally being used perhaps on a later page.

```
    \let\LWR@newautopagelabel\LWR@null@newautopagelabel%
  \ignorespaces%
}%
```

`\V@@@footnotetext`

```
\def\V@@@footnotetext{%
    \LWR@origtilde\LWR@orignewline%
    \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
    \strut\egroup%
}
}% AfterEndPreamble

\renewcommand*{\@shadowbox}[1]{%
\ifbool{FormatWP}%
{\InlineClass[border:1px solid black]{shadowbox}{#1}}%
{\InlineClass{shadowbox}{#1}}%
}

\renewcommand*{\@doublebox}[1]{%
\ifbool{FormatWP}%
{\InlineClass[border:1px double black]{doublebox}{#1}}%
{\InlineClass{doublebox}{#1}}%
}

\renewcommand*{\@ovalbox}[2]{%
\ifbool{FormatWP}%
{\InlineClass[border:1px solid black; border-radius:1ex]{ovalbox}{#2}}%
{%
    \ifthenelse{\isequivalentto{#1}{\thinlines}}%
        {\InlineClass{ovalbox}{#2}}%
        {\InlineClass{Ovalbox}{#2}}%
}%
}
```

Convert minipages, parboxes, and lists into linear text using the `LWR@nestspan` environment:

```
\let\LWR@origSbox\Sbox

\def\Sbox{\LWR@origSbox\LWR@nestspan}


\let\LWR@origendSbox\endSbox

\def\endSbox{\endLWR@nestspan\LWR@origendSbox}
```

`Beqnarray` is adapted for MathJax or enclosed inside a `lateximage`:

```
\RenewEnviron{Beqnarray}
{\LWR@eqnarrayfactor}

\csgpreto{Beqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

`\GenericCaption` is enclosed in an HTML block:

```
\renewcommand{\GenericCaption}[1]{%
    \LWR@figcaption%
    \LWR@isolate{#1}%
    \endLWR@figcaption%
}
```

`Btrivlist` is enclosed in an HTML block. This is a tabular, and does not use `\item`.

`\trivlist` {⟨*l/c/r*⟩} [⟨*t/c/b*⟩]

```
\RenewDocumentEnvironment{Btrivlist}{m o}
{%
    \LWR@stoppars%
    \begin{BlockClass}{Btrivlist}%
    \tabular{#1}%
}
{%
    \endtabular%
    \end{BlockClass}%
    \LWR@startpars%
}
```

`Btrivlist` is also neutralized when used inside a span:

```
\AtBeginEnvironment{LWR@nestspan}{%
    \RenewDocumentEnvironment{Btrivlist}{m o}{}{}%
}
```

lwarp's handling of `\item` is patched to accept fancybox's optional arguments:

```
\let\LWRFB@origitemizeitem\LWR@itemizeitem
\let\LWRFB@origdescitem\LWR@descitem

\RenewDocumentCommand{\LWR@itemizeitem}{d()o}{%
    \IfValueTF{#2}{%
        \LWRFB@origitemizeitem[#2]%
    }{%
        \LWRFB@origitemizeitem%
    }%
}

\RenewDocumentCommand{\LWR@descitem}{d()o}{%
    \IfValueTF{#2}{%
        \LWRFB@origdescitem[#2]~%
    }{%
        \LWRFB@origdescitem%
    }%
}
```

```
\RenewDocumentCommand{\LWR@nestspanitem}{d()}{%
    \if@newlist\else{
        \LWR@htmltagc{br /}%
        \LWR@orignewline%
    }\fi%
    \LWR@origitem%
}
```

The various boxed lists become regular lists:

```
\renewenvironment{Bitemize}[1][]
    {%
        \LWR@spanwarnformat{Bitemize}%
        \booltrue{LWR@starting@fancybox}%
        \begin{itemize}%
        \boolfalse{LWR@starting@fancybox}%
    }
    {\end{itemize}}

\renewenvironment{Benumerate}[1][]
    {%
        \LWR@spanwarnformat{Benumerate}%
        \booltrue{LWR@starting@fancybox}%
        \begin{enumerate}%
        \boolfalse{LWR@starting@fancybox}%
    }
    {\end{enumerate}}

\renewenvironment{Bdescription}[1][]
    {%
        \LWR@spanwarnformat{Bdescription}%
        \booltrue{LWR@starting@fancybox}%
        \begin{description}%
        \boolfalse{LWR@starting@fancybox}%
    }
    {\end{description}}
```

\boxput simply prints one then the other argument, side-by-side instead of above and behind:

```
\RenewDocumentCommand{\boxput}{s d() m m}{%
    \IfBooleanTF{#1}{#3\quad#4}{#4\quad#3}%
}
```

Neutralized commands:

```
\RenewDocumentCommand{\fancyput}{s d() m}{}
\RenewDocumentCommand{\thisfancyput}{s d() m}{}

\RenewDocumentCommand{\fancypage}{m m}{}
\RenewDocumentCommand{\thisfancypage}{m m}{}

\def\LandScape#1{}
\def\endLandScape{}
\def\@Landscape#1#2#3{}
\def\endLandscape{}
```

Low-level patches for UseVerbatim and friends:

```
\let\LWRFB@UseVerbatim\UseVerbatim
\renewcommand*{\UseVerbatim}[1]{%
    \LWR@atbeginverbatim{Verbatim}%
    \LWRFB@UseVerbatim{#1}%
    \LWR@afterendverbatim%
}

\let\LWRFB@LUseVerbatim\LUseVerbatim

\renewcommand*{\LUseVerbatim}[1]{%
    \LWR@atbeginverbatim{LVerbatim}%
    \noindent%
    \LWRFB@LUseVerbatim{#1}%
    \LWR@afterendverbatim%
}

\def\@BUseVerbatim[#1]#2{%
    \LWR@atbeginverbatim{BVerbatim}%
    \LWRFB@UseVerbatim{#2}%
    \LWR@afterendverbatim%
}
```

File 151 **lwarp-fancyhdr.sty**

§ 260 Package

fancyhdr

(*Emulates or patches code by* Piet van Oostrum.)

fancyhdr (*Pkg*) fancyhdr is ignored.

for HTML output: Discard all options for lwarp-fancyhdr:

```
\LWR@ProvidesPackageDrop{fancyhdr}[2021/01/04]

\newcommand*{\fancyhead}[2][]{}
\newcommand*{\fancyfoot}[2][]{}
\newcommand*{\fancyhf}[2][]{}

\newcommand*{\lhead}[2][]{}
\newcommand*{\chead}[2][]{}
\newcommand*{\rhead}[2][]{}
\newcommand*{\lfoot}[2][]{}
\newcommand*{\cfoot}[2][]{}
\newcommand*{\rfoot}[2][]{}
\newcommand*{\headrulewidth}{}
\newcommand*{\footrulewidth}{}
\providecommand{\headruleskip}{0pt}
\providecommand{\footruleskip}{0pt}
\newcommand{\plainheadrulewidth}{0pt}
\newcommand{\plainfootrulewidth}{0pt}
\def\fancyplain#1#2{#1}
\newcommand*{\headrule}{}
\newcommand*{\footrule}{}
\newlength{\headwidth}
\newcommand*{\fancycenter}[1][1em]{}
```

```
\newcommand*{\fancyheadoffset}[2][]{}
\newcommand*{\fancyfootoffset}[2][]{}
\newcommand*{\fancyhfoffset}[2][]{}
\newcommand{\fancyheadinit}[1]{}
\newcommand{\fancyfootinit}[1]{}
\newcommand{\fancyhfinit}[1]{}
\newcommand*{\iffloatpage}[2]{#2}
\newcommand*{\ifftopfloat}[2]{#2}
\newcommand*{\iffbotfloat}[2]{#2}
\newcommand*{\iffootnote}[2]{#2}

\newcommand{\fancypagestyle}[1]{%
  \@ifnextchar[{\f@nch@pagestyle{#1}}{\f@nch@pagestyle{#1}[]}%
}
\long\def\f@nch@pagestyle#1[#2]#3{}
```

File 152 **lwarp-fancypar.sty**

§ 261 Package

fancypar

(*Emulates or patches code by* Gonzalo Medina.)

`fancypar` (*Pkg*) fancypar is used as-is for print output, and emulated for HTML.

⚠ css classes `\NotebookPar` and related are used as-is inside a `lateximage`, but for HTML these are emulated as a `<div>` of class `NotebookPar`, etc. For HTML, the package options and the macro optional arguments are ignored. The user must provide custom CSS for each if visual effects are required. See section 7.7.

⚠ custom styles If using a custom paragraph style, such as `\MyStylePar` from the documentation, use the following to generate an HTML `<div>` of class `MyStylePar`:

```
... (existing definiton of \MyStylePar, print version) ...
\begin{warpHTML}
\AddFancyparClass{MyStyle}
\end{warpHTML}
```

`\MyStylePar` is then modified to emulate HTML. An optional argument is allowed, which is ignored.

for HTML output:

```
\LWR@ProvidesPackagePass{fancypar}[2019/01/18]

\begin{warpHTML}
\makeatletter

\newcommand{\LWR@fancypar}[2]{%
    \begin{BlockClass}{#1Par}
    #2
    \end{BlockClass}
}

\newcommand{\LWR@HTML@NotebookPar}[2][]{\LWR@fancypar{Notebook}{#2}}
\LWR@formatted{NotebookPar}

\newcommand{\LWR@HTML@ZebraPar}[2][]{\LWR@fancypar{Zebra}{#2}}
\LWR@formatted{ZebraPar}
```

```
16
17 \newcommand{\LWR@HTML@DashedPar}[2][]{\LWR@fancypar{Dashed}{#2}}
18 \LWR@formatted{DashedPar}
19
20 \newcommand{\LWR@HTML@MarkedPar}[2][]{\LWR@fancypar{Marked}{#2}}
21 \LWR@formatted{MarkedPar}
22
23 \newcommand{\LWR@HTML@UnderlinedPar}[2][]{\LWR@fancypar{Underlined}{#2}}
24 \LWR@formatted{UnderlinedPar}
25
26
27 \newcommand{\LWR@HTML@add@fancy@format}{}
28 \LWR@formatted{add@fancy@format}
29
30
31 \newcommand{\AddFancyparClass}[1]{%
32     \expandafter\newcommand\csname LWR@HTML@#1Par\endcsname[2][]{%
33         \LWR@fancypar{#1}{##2}%
34     }
35     \LWR@formatted{#1Par}
36 }
37
38 \makeatother
39 \end{warpHTML}
```

File 153 **lwarp-fancyref.sty**

§ 262 Package fancyref

(*Emulates or patches code by* Axel Reichert.)

`fancyref` (*Pkg*) fancyref is modifed for HTML output.

for HTML output:

```
1 \LWR@ProvidesPackagePass{fancyref}[1999/02/03]
```

`\fancyrefhook` (*Hook*) [fancyref] To remove the `margin` option, if `\fancyrefhook` is anything other than the `paren` option, then force it to the default instead. (Comparing to the `margin` option was not possible since lwarp has revised the meaning of `\mbox` so the comparison failed.)

```
2 \newcommand*{\LWRfref@parenfancyrefhook}[1]{(#1)}
3
4 \ifdefstrequal{\fancyrefhook}{\LWRfref@parenfancyrefhook}
5 {}{
6     \renewcommand*{\fancyrefhook}[1]{#1}%
7 }
```

File 154 **lwarp-fancytabs.sty**

§ 263 Package fancytabs

`fancytabs` (*Pkg*) fancytabs is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{fancytabs}[2016/03/29]
```

```
\newcommand{\fancytab}[3][RIGHT]{}
\newcommand{\fancytabsStyle}[1]{}
\newcommand{\fancytabsHeight}[1]{}
\newcommand{\fancytabsWidth}[1]{}
\newcommand{\fancytabsCount}[1]{}
\newcommand{\fancytabsLeftColor}[1]{}
\newcommand{\fancytabsRightColor}[1]{}
\newcommand{\fancytabsTop}[1]{}
\newcommand{\fancytabsTextVPos}[1]{}
\newcommand{\fancytabsTextHPos}[1]{}
\newcommand{\fancytabsGap}[1]{}
\newcommand{\fancytabsFloor}[1]{}
\newcommand{\fancytabsRotate}[1]{}
```

File 155 **lwarp-fancyvrb.sty**

§ 264 Package

fancyvrb

(*Emulates or patches code by* Timothy Van Zandt.)

fancyvrb (*Pkg*)

fancyvrb is supported with some patches.

HTML classes

The fancy verbatim environment is placed inside a of class fancyvrb. The label is placed inside a of class fancyvrblabel. The verbatim text itself is placed inside a of class verbatim.

fancybox, fancyvrb

⚠ \VerbatimFootnotes

⚠ sectioning or displaymath

If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

At present there is a bug such that paragraph closing tags are not present in footnotes when \VerbatimFootnotes are selected. The browser usually compensates.

```
\AtBeginDocument{\RequirePackage{xcolor}}% for \convertcolorspec

\LWR@ProvidesPackagePass{fancyvrb}[2008/02/07]
```

Initial default patch for fancyvrb:

```
\fvset{frame=none}%
```

After the preamble is loaded, after any patches to Verbatim:

```
\AfterEndPreamble{
\LWR@traceinfo{Patching fancyvrb.}
```

\VerbatimFootnotes

Patched to use the new version.

```
\def\VerbatimFootnotes{%
    \let\@footnotetext\V@footnotetext%
```

```
    \let\footnote\V@footnote%
    \let\LWR@footnotetext\V@footnotetext% lwarp
}
```

\V@@footnotetext

Patches in a subset of lwarp's \LWR@footnotetext to the fancyvrb version of \V@@footnotetext.

```
\def\V@@footnotetext{%
\LWR@traceinfo{V@footnotetext}%
```

Place an autopage marker so that back references to citations inside a footnote will link closer to the footnote text, if possible.

```
    \LWR@newautopagelabel{page}%
```

Take the current footnote box, then append:

```
    \global\setbox\LWR@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
    \unvbox\LWR@footnotebox%
```

Remember the footnote number for \ref:

```
    \protected@edef\@currentlabel{%
        \csname p@footnote\endcsname\@thefnmark%
    }% @currentlabel
```

Use HTML superscripts in the footnote even inside a lateximage:

```
    \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
    \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Append the footnote mark to the list:

```
    \@makefntext{}%
```

The footnote text will follow after \V@@@footnotetext has completed.

```
    \bgroup%
    \aftergroup\V@@@footnotetext%
```

Do not generate autopages inside the footnotes, since they are accumulated at the moment before finally being used perhaps on a later page.

```
    \let\LWR@newautopagelabel\LWR@null@newautopagelabel%

    \ignorespaces%
}%
```

\V@@@footnotetext

```
\def\V@@@footnotetext{%
    \LWR@origtilde\LWR@orignewline%
    \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
    \strut\egroup%
}

\preto\FVB@Verbatim{\LWR@forcenewpage}
\preto\FVB@LVerbatim{\LWR@forcenewpage}
% \preto\FVB@BVerbatim{\LWR@forcenewpage}% Fails, so done below.
```

Simplified to remove PDF formatting:

```
\def\FV@BeginListFrame@Single{%
  \FV@SingleFrameLine{\z@}%
}

\def\FV@EndListFrame@Single{%
  \FV@SingleFrameLine{\@ne}%
}

\def\FV@BeginListFrame@Lines{%
  \FV@SingleFrameLine{\z@}%
}

\def\FV@EndListFrame@Lines{%
    \FV@SingleFrameLine{\@ne}%
}

\renewcommand*{\FV@SingleFrameSep}{}
```

Adds HTML formatting:

```
\def\FV@BUseVerbatim#1{%
    \FV@BVerbatimBegin#1\FV@BVerbatimEnd%
}
```

`\LWR@FVstyle` Holds the style of the verbatim.

```
\newcommand*{\LWR@FVstyle}{}
```

The following patches to `Verbatim` are executed at the start and end of the environment, depending on the choice of `frame`. Original code is from the fancyvrb package.

```
\newcommand*{\LWR@fvstartnone}{%
\LWR@traceinfo{fvstartnone}%
% \hbox to\z@{
\BlockClass[\LWR@FVstyle]{fancyvrb}
\LWR@stoppars
\ifx\FV@LabelPositionTopLine\relax\else
    \ifx\FV@LabelBegin\relax\else
        \FancyVerbRuleColor{\LWR@FVfindbordercolor}
        \LWR@htmltagc{%
            div class=\textquotedbl{}fancyvrblabel\textquotedbl\ % space
           style=\textquotedbl{}color: \LWR@origpound\LWR@tempcolor\textquotedbl%
        }
        \LWR@print@textrm{\FV@LabelBegin}% \textrm preserves emdash
        \LWR@htmltagc{/div}\LWR@orignewline%
    \fi
\fi
\LWR@atbeginverbatim{verbatim}%
% }%
}

\newcommand*{\LWR@fvendnone}{%
\LWR@traceinfo{fvendnone}%
% \hbox to\z@{
\LWR@afterendverbatim%
```

```
\LWR@stoppars%
\ifx\FV@LabelPositionBottomLine\relax\else
    \ifx\FV@LabelEnd\relax\else
        \FancyVerbRuleColor{\LWR@FVfindbordercolor}
        \LWR@htmltagc{%
            div class=\textquotedbl{}fancyvrblabel\textquotedbl\ % space
           style=\textquotedbl{}color: \LWR@origpound\LWR@tempcolor\textquotedbl%
        }
        \LWR@print@textrm{\FV@LabelEnd}
        \LWR@htmltagc{/div}\LWR@orignewline%
    \fi
\fi
\endBlockClass
}

\newcommand*{\LWR@fvstartsingle}{%
\LWR@traceinfo{fvstartsingle}%
\LWR@fvstartnone%
\FV@BeginListFrame@Single%
}

\newcommand*{\LWR@fvendsingle}{%
\LWR@traceinfo{fvendsingle}%
\FV@EndListFrame@Single%
\LWR@fvendnone%
}

\newcommand*{\LWR@fvstartline}{%
\LWR@traceinfo{fvstartline}%
\LWR@fvstartnone%
% \setlength{\LWR@templengthone}{\baselineskip}%
\FV@BeginListFrame@Lines%
% \setlength{\baselineskip}{\LWR@templengthone}%
% \setlength{\baselineskip}{5pt}%
}

\newcommand*{\LWR@fvendline}{%
\LWR@traceinfo{fvendline}%
\FV@EndListFrame@Lines%
\LWR@fvendnone%
}
```

The following patches select the start/left/right/end behaviors depending on `frame`. Original code is from the fancyvrb package.

```
\newcommand*{\LWR@FVfindbordercolor}{%
\FancyVerbRuleColor%
\LWR@findcurrenttextcolor%
\color{black}%
}

% border width of \FV@FrameRule
\newcommand*{\LWR@FVborderstyle}[1]{%
padding#1: \strip@pt\dimexpr \FV@FrameSep\relax\relax pt ; % space
\LWR@FVfindbordercolor\LWR@indentHTMLtwo%
border#1: \strip@pt\dimexpr \FV@FrameRule\relax\relax pt % space
solid {\FancyVerbRuleColor{\LWR@origpound\LWR@tempcolor}} ; % space
}

\def\FV@Frame@none{%
```

```
\renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle}%
\let\FV@BeginListFrame\LWR@fvstartnone%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendnone}

\FV@Frame@none% default values

\def\FV@Frame@single{%
\renewcommand*{\LWR@FVstyle}{%
    \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
    \LWR@FVborderstyle{}%
}%
\let\FV@BeginListFrame\LWR@fvstartsingle%
\let\FV@LeftListFrame\FV@LeftListFrame@Single%
\let\FV@RightListFrame\FV@RightListFrame@Single%
\let\FV@EndListFrame\LWR@fvendsingle}

\def\FV@Frame@lines{%
\renewcommand*{\LWR@FVstyle}{%
    \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
    \LWR@FVborderstyle{-top}%
    \LWR@indentHTMLtwo%
    \LWR@FVborderstyle{-bottom}%
}%
\let\FV@BeginListFrame\LWR@fvstartline%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendline}

\def\FV@Frame@topline{%
\renewcommand*{\LWR@FVstyle}{%
    \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
    \LWR@FVborderstyle{-top}%
}%
\let\FV@BeginListFrame\LWR@fvstartline%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendnone}

\def\FV@Frame@bottomline{%
\renewcommand*{\LWR@FVstyle}{%
    \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
    \LWR@FVborderstyle{-bottom}%
}%
\let\FV@BeginListFrame\LWR@fvstartnone%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendline}
```

Seems to be required in some situations:

```
\def\FV@FrameFillLine{}%
```

```
\def\FV@Frame@leftline{%
\renewcommand*{\LWR@FVstyle}{%
    \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
    \LWR@FVborderstyle{-left}%
}%
% To define the \FV@FrameFillLine macro (from \FV@BeginListFrame)
```

```
\ifx\FancyVerbFillColor\relax%
\let\FV@FrameFillLine\relax%
\else%
\@tempdima\FV@FrameRule\relax%
\multiply\@tempdima-\tw@%
\edef\FV@FrameFillLine{%
{\noexpand\FancyVerbFillColor{\vrule\@width\number\@tempdima sp}%
\kern-\number\@tempdima sp}}%
\fi%
\let\FV@BeginListFrame\LWR@fvstartnone%
\let\FV@LeftListFrame\FV@LeftListFrame@Single%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendnone}
```

Adds the optional label to the top and bottom edges. Original code is from the fancyvrb package.

```
\def\FV@SingleFrameLine#1{%
%   \hbox to\z@{%
%     \kern\leftmargin
    \ifnum#1=\z@\relax
      \let\FV@Label\FV@LabelBegin
    \else
      \let\FV@Label\FV@LabelEnd
    \fi
    \ifx\FV@Label\relax
%       \FancyVerbRuleColor{\vrule \@width\linewidth \@height\FV@FrameRule}%
    \else
      \ifnum#1=\z@
%         \setbox\z@\hbox{\strut\enspace\FV@LabelBegin\enspace\strut}%
        \ifx\FV@LabelPositionTopLine\relax
        \else
        \fi
      \else
%         \setbox\z@\hbox{\strut\enspace\FV@LabelEnd\enspace\strut}%
        \ifx\FV@LabelPositionBottomLine\relax
        \else
        \fi
      \fi
    \fi
%     \hss
%     }
}
```

Processes each line, adding optional line numbers. Original code is from the fancyvrb package.

```
\def\FV@ListProcessLine#1{%
   \hbox to \hsize{%
%      \kern\leftmargin
      \hbox to \VerbatimHTMLWidth {%
       \ifcsvoid{FV@LeftListNumber}{}{\kern 2.5em}%
        \FV@LeftListNumber%
%       \FV@LeftListFrame
      \FancyVerbFormatLine{#1}%
      \hss%
%       \FV@RightListFrame
      \FV@RightListNumber%
    }%
```

```
      \hss% required to avoid underfull hboxes
}
}

\def\FV@ListProcessLine@i#1{%
%   \hbox{%
    \ifvoid\@labels\else
      \hbox to \z@{\kern\@totalleftmargin\box\@labels\hss}%
    \fi
    \FV@ListProcessLine{#1}%
%   }%
%   \let\FV@ProcessLine\FV@ListProcessLine@ii%
}

\def\FV@ListProcessLastLine{}
```

BVerbatim (*env.*)

```

\xpretocmd{\FV@BeginVBox}
    {%
        \LWR@forcenewpage% instead of \preto
        \LWR@atbeginverbatim{bverbatim}%
    }
    {}
    {\LWR@patcherror{fancyvrb}{FV@BeginVBox}}

\xapptocmd{\FV@EndVBox}
    {%
        \LWR@afterendverbatim%
    }
    {}
    {\LWR@patcherror{fancyvrb}{FV@EndVBox}}
```

End of the modifications to make at the end of the preamble:

```
} % \AfterEndPreamble
```

File 156 **lwarp-fbox.sty**

§ 265 Package

fbox

(*Emulates or patches code by* Herbert Voss.)

fbox (*Pkg*) fbox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{fbox}[2022/02/20]
```

This will be \LWR@formatted when \AtBeginDocument:

```
\LetLtxMacro\LWR@HTML@fbox\fbox
```

Instead of using the original, the new version is used with all borders:

```
\renewcommand*{\orig@fbox}{\FBox@i[tblr]}
```

\WR@fboxpkg@border {⟨*1: top/bottom/left/right*⟩} {⟨*2: t/b/l/r*⟩} {⟨*3: padding, or empty*⟩}

Accumulates HTML styles for border, and padding if given:

```
\newcommand*{\LWR@fboxpkg@border}[3]{%
    \colorlet{LWR@border@color}{\csuse{fbox@#2color}}%
  \protect\convertcolorspec{named}{LWR@border@color}{HTML}\LWR@tempbordercolor\relax%
    \appto\LWR@tempone{%
        border-#1: % space
        \LWR@printlength{\LWR@atleastonept} % space
        solid \LWR@origpound%
    }%
  \expandafter\appto\expandafter\LWR@tempone\expandafter{\LWR@tempbordercolor}%
    \appto\LWR@tempone{ ;\LWR@indentHTML}%
    \ifblank{#3}{}{%
        \appto\LWR@tempone{%
            padding-#1: \LWR@printlength{#3} ;\LWR@indentHTML
        }%
    }%
}
```

A hack to reuse the same code for inline and blocks:

```
\newbool{LWR@fboxpkg@ispar}
\boolfalse{LWR@fboxpkg@ispar}
```

Acculumate HTML styles for left and right padding, depending on `\if@fbox@space@left`, `\if@fbox@space@right`:

```
\newcommand{\LWR@fboxpkg@lrpadding}[1]{%
    \csuse{if@fbox@space@#1}%
        \appto\LWR@tempone{%
            padding-#1: \LWR@printlength{\fbox@@sep};\LWR@indentHTML
        }
    \else%
        \appto\LWR@tempone{%
            padding-#1: 0pt;\LWR@indentHTML
        }
    \fi%
}
```

The HTML version, modified to use HTML styles and either an `\InlineClass` or `BlockClass`:

```
\newcommand{\LWR@HTML@FBox@iii}[1]{%
```

Find and set the text color, rule width, margin:

```
    \LWR@forceminwidth{\fbox@@rule}%
    \LWR@findcurrenttextcolor%
    \def\LWR@tempone{%
        color: \LWR@origpound\LWR@tempcolor ; \LWR@indentHTML
        margin: 1ex ; \LWR@indentHTML
    }%
```

Add left/right padding:

```
    \LWR@fboxpkg@lrpadding{left}%
    \LWR@fboxpkg@lrpadding{right}%
```

Per the original to decode the borders, in a new way:

```
    \ifnum\the\@tempcntb>8\relax
        \advance\@tempcntb by -8\relax
        \LWR@fboxpkg@border{top}{t}{\fbox@@sep}%
    \fi
    \ifnum\@tempcntb>3
        \advance\@tempcntb by -4\relax
        \LWR@fboxpkg@border{left}{l}{}%
    \fi
    \ifnum\@tempcntb>1\relax
        \LWR@fboxpkg@border{right}{r}{}%
    \fi
    \ifodd\@tempcntb
        \LWR@fboxpkg@border{bottom}{b}{\fbox@@sep}%
    \fi
```

Generate a `BlockClass` or `\InlineClass` with the contents:

```
    \color@begingroup
    \ifbool{LWR@fboxpkg@ispar}%
        {%
            \begin{BlockClass}[\LWR@tempone]{fboxpkg}%
                #1%
            \end{BlockClass}%
        }%
        {%
            \InlineClass[\LWR@tempone]{fboxpkg}{%
                #1%
            }%
        }%
    \color@endgroup
    \boolfalse{LWR@fboxpkg@ispar}% globally
}
\LWR@formatted{FBox@iii}
```

For `\fparbox`, set the use of `BlockClass`, then reuse the above:

```
\long\def\LWR@HTML@FParBox@i[#1]#2{%
    \booltrue{LWR@fboxpkg@ispar}%
    \FBox@i[#1]{#2}
}
\LWR@formatted{FParBox@i}

\long\def\LWR@HTML@FParBox@ii#1{%
    \booltrue{LWR@fboxpkg@ispar}%
    \FBox@i[tblr]{#1}%
}
\LWR@formatted{FParBox@ii}
```

For MathJax, absorb and ignore star and optional arguments:

```
\CustomizeMathJax{\let\LWRorigfbox\fbox}
\CustomizeMathJax{\newcommand{\LWRfboxpkgtwo}[2][]{\LWRorigfbox{#2}}}
\CustomizeMathJax{\renewcommand{\fbox}{\ifstar\LWRfboxpkgtwo\LWRfboxpkgtwo}}
\CustomizeMathJax{\newcommand{\fparbox}{\fbox}}
```

File 157 **lwarp-fewerfloatpages.sty**

§ 266 Package

fewerfloatpages

fewerfloatpages (*Pkg*) fewerfloatpages is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fewerfloatpages}[2020/02/14]

\newcommand\floatpagekeepfraction{\textfraction}
\newcounter{floatpagedeferlimit}
\newcounter{floatpagekeeplimit}
```

File 158 **lwarp-figcaps.sty**

§ 267 Package

figcaps

(*Emulates or patches code by* Patrick W. Daly.)

figcaps (*Pkg*) figcaps is ignored.

for HTML output: Discard all options for lwarp-figcaps:

```
\LWR@ProvidesPackageDrop{figcaps}[1999/02/23]

\newcommand*{\figcapson}{}
\newcommand*{\figcapsoff}{}
\newcommand*{\printfigures}{}
\newcommand*{\figmarkon}{}
\newcommand*{\figmarkoff}{}
\def\figurecapname{Figure Captions}
\def\tablepagename{Tables}
\def\figurepagename{Figures}
```

File 159 **lwarp-figsize.sty**

§ 268 Package

figsize

(*Emulates or patches code by* Anthony A. Tanbakuchi.)

figsize (*Pkg*) figsize is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{figsize}[2002/03/18]
```

Emulates a virtual 6×9 inch textsize.

```
\newlength{\figwidth}
\newlength{\figheight}

\newcommand{\SetFigLayout}[3][0]{%
```

```
\setlength{\figheight}{8in}%
\setlength{\figheight}{\figheight / #2}%
%
\setlength{\figwidth}{5.5in}%
\setlength{\figwidth}{\figwidth / #3}%
}
```

File 160 **lwarp-fitbox.sty**

§ 269 Package

fitbox

`fitbox` (*Pkg*) fitbox is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fitbox}[2019/02/20]

\NewDocumentCommand{\fitbox}{s o m}{%
    \begin{BlockClass}{fitbox}
    #3
    \end{BlockClass}
}

\newcommand*{\fitboxset}[1]{}

\newdimen\fitboxnatheight
\newdimen\fitboxnatwidth

\newcommand\SetFitboxLayout[3][]{}
```

File 161 **lwarp-fix2col.sty**

§ 270 Package

fix2col

`fix2col` (*Pkg*) fix2col is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fix2col}[2015/11/13]
```

File 162 **lwarp-fixmath.sty**

§ 271 Package

fixmath

(*Emulates or patches code by* Walter Schmidt.)

`fixmath` (*Pkg*) fixmath is used as-is for SVG math, and emulated for MathJax.

⚠ limitations MathJax does not have full font support for bold italic Greek.

for HTML output:

```
\LWR@ProvidesPackagePass{fixmath}[2000/04/11]

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
```

```
\LWR@mathjax@addgreek@u@it*{}{}
\LWR@mathjax@addletter{\BooleanTrue}{up}{}{delta}{0394}
\LWR@mathjax@addletter{\BooleanTrue}{up}{}{omega}{03A9}
\CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
\end{warpMathJax}
```

File 163 **lwarp-fixme.sty**

§ 272 Package

fixme

(*Emulates or patches code by* Didier Verna.)

`fixme` (*Pkg*) fixme is patched for use by lwarp.

⚠ external layouts External layouts (`\fxloadlayouts`) are not supported.

Customized layouts are overwritten by lwarp's versions `\AtBeginDocument` in order to provide the HTML conversion. If creating a new layout, see lwarp's changes to provide similar for the new layout, inside a `warpHTML` environment.

User control is provided for setting the HTML styling of the "faces". The defaults are as follows, and may be changed in the preamble after fixme is loaded:

```
\def\FXFaceInlineHTMLStyle{font-weight:bold}
\def\FXFaceEnvHTMLStyle{font-weight:bold}
\def\FXFaceSignatureHTMLStyle{font-style:italic}
\def\FXFaceTargetHTMLStyle{font-style:italic}
```

for HTML output:

```
\LWR@ProvidesPackagePass{fixme}[2019/01/03]
```

Restore lwarp's version of `\@wrindex`, ignoring the fixme package's `target` option:

```
\let\@wrindex\LWR@wrindex
```

Float-related macros required by lwarp:

```
\newcommand{\ext@fixme}{lox}

\renewcommand{\l@fixme}[2]{%
    \hypertocfloat{1}{fixme}{lox}%
     {\LWR@nameref{\BaseJobname-autopage-\arabic{LWR@nextautopage}} --- #1}%
        {#2}
}
```

Other modifications. Done `\AtBeginDocument` to hopefully work if the user customizes the layouts.

```
\AtBeginDocument{

\def\FXFaceInlineHTMLStyle{font-weight:bold}

\renewcommand*\FXLayoutInline[3]{ % space
    \InlineClass[\FXFaceInlineHTMLStyle]{fixmeinline}%
        {\@fxtextstd{#1}{#2}{#3}}%
}

```

```
\def\FXFaceEnvHTMLStyle{font-weight:bold}

\renewcommand*\FXEnvLayoutPlainBegin[2]{%
    \BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
    \ignorespaces#2 \fxnotename{#1}: \ignorespaces%
}

\renewcommand*\FXEnvLayoutPlainEnd[2]{\endBlockClass}

\renewcommand*\FXEnvLayoutSignatureBegin[2]{%
    \BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
    \fxnotename{#1}: \ignorespaces%
}

\renewcommand*\FXEnvLayoutSignatureEnd[2]{\@fxsignature{#2}\endBlockClass}

\def\FXFaceSignatureHTMLStyle{font-style:italic}

\DeclareRobustCommand*\@fxsignature[1]{%
    \ifthenelse{\equal{#1}{}}%
        {}%
        { -- {\InlineClass[\FXFaceSignatureHTMLStyle]{fixmesignature}{#1}}}%
}


\def\FXFaceTargetHTMLStyle{font-style:italic}

\renewcommand\FXTargetLayoutPlain[2]{%
    \InlineClass[\FXFaceTargetHTMLStyle]{fixmetarget}{#2}%
}

}% \AtBeginDocument
```

File 164 **lwarp-fixmetodonotes.sty**

§ 273 Package

fixmetodonotes

(*Emulates or patches code by* Gioele Barabucci.)

`fixmetodonotes` (*Pkg*) fixmetodonotes is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{fixmetodonotes}[2013/04/28]

\renewcommand{\NOTES@addtolist}[2]{%
    \refstepcounter{NOTES@note}%
%   \phantomsection% REMOVED
    \addcontentsline{notes}{NOTES@note}{%
        \protect\numberline{\theNOTES@note}{{#1}: {#2}}%
    }%
}

\renewcommand{\NOTES@marker}[2]{\fbox{%
    \textcolor{#2}{% WAS \color
        \textbf{#1}}%
    }}

\renewcommand{\NOTES@colorline}[2]{%
```

```
16  \bgroup%
17    \ULon{\LWR@backgroundcolor{#1}{#2}}%
18 }
```

File 165 **lwarp-flafter.sty**

§ 274 Package

flafter

`flafter` (*Pkg*) flafter is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{flafter}[2018/01/08]
2 \providecommand\fl@trace[1]{}
```

File 166 **lwarp-flippdf.sty**

§ 275 Package

flippdf

`flippdf` (*Pkg*) flippdf is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{flippdf}[2006/06/30]

2 \newcommand\FlipPDF{}
3 \newcommand\UnFlipPDF{}
```

File 167 **lwarp-float.sty**

§ 276 Package

float

(*Emulates or patches code by* Anselm Lingnau.)

`float` (*Pkg*) float is emulated.

Float styles `boxed` and `ruled` are emulated by CSS and a float class according to style.

The HTML `<figure>` `class` is set to the float type, so CSS may also be used to format the float and its caption, according to float type. Furthermore, an additional class is set to the float style: `plain`, `plaintop`, `boxed`, or `ruled`, so CSS may be used to format by float style as well. Default formatting by CSS is provided for `ruled` and `boxed` styles.

⚠ not seem to be a floating environment

Always declare a `\newfloat` before modifying it with `\floatname`, etc.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{float}[2001/11/08]
```

`\LWR@floatstyle` The default float style.

```
2 \newcommand*{\LWR@floatstyle}{plain}
```

`\newfloat` {⟨*1: type*⟩} {⟨*2: placement*⟩} {⟨*3: ext*⟩} [⟨*4: within*⟩]

Emulates the `\newfloat` command from the float package.

"placement" is ignored.

```
\NewDocumentCommand{\newfloat}{m m m o}{%
    \IfValueTF{#4}%
        {\DeclareFloatingEnvironment[fileext=#3,within=#4]{#1}}%
        {\DeclareFloatingEnvironment[fileext=#3]{#1}}%
```

Remember the float style:

```
    \csedef{LWR@floatstyle@#1}{\LWR@floatstyle}%
```

newfloat package automatically creates the `\listof` command for new floats, but float does not, so remove `\listof` here in case it is manually created later.

```
    \cslet{listof#1s}\relax%
    \cslet{listof#1es}\relax%
```

Likesize, newfloat also creates `\l@<type>`, but float does not, so remove it here:

```
    \cslet{l@#1}\relax%
}
```

`\floatname` {⟨*type*⟩} {⟨*name*⟩}

Sets the text name of the float, such as "Figure". Avoids trying to set a recursive name, from trivfloat.

```
\NewDocumentCommand{\floatname}{m +m}{%
    \def\LWR@tempone{#2}%
    \def\LWR@temptwo{\@nameuse{#1name}}%
    \ifdefequal{\LWR@tempone}{\LWR@temptwo}{}{%
        \SetupFloatingEnvironment{#1}{name=#2}%
    }%
}
```

`\floatplacement` {⟨*type*⟩} {⟨*placement*⟩}

Float placement is ignored.

```
\newcommand*{\floatplacement}[2]{%
    \SetupFloatingEnvironment{#1}{placement=#2}%
}
```

`\floatstyle` {⟨*style*⟩}

Remember the style for future floats:

```
\newcommand{\floatstyle}[1]{%
    \def\LWR@floatstyle{#1}%
}%
```

`\restylefloat` * {⟨*type*⟩}

Remember the style for this float:

```
\NewDocumentCommand{\restylefloat}{s m}{%
    \csedef{LWR@floatstyle@#2}{\LWR@floatstyle}%
}
```

\listof

See section 78.2 for the \LWR@listof command in the lwarp core.

```
\newcommand{\listof}{\LWR@listof}
```

File 168 **lwarp-floatflt.sty**

§ 277 Package

floatflt

(*Emulates or patches code by* Mats Dahlgren.)

floatflt (*Pkg*)

floatflt is emulated.

for HTML output:

Discard all options for lwarp-floatflt:

```
\LWR@ProvidesPackageDrop{floatflt}[1997/07/16]
```

Env [⟨⟩]

offset {⟨*type*⟩} {⟨*width*⟩} Borrowed from the lwarp version of keyfloat:

```
\NewDocumentEnvironment{KFLTfloatflt@marginfloat}{O{-1.2ex} m m}
{%
    \begin{LWR@setvirtualpage}*%
    \ifblank{#3}{%
        \LWR@BlockClassWP{%
            float:right; %
            width: 1.5in; % reasonable dummy width for word processor
            margin:10pt%
        }{}%
        (note)%
        {marginblock}%
    }{%
        \setlength{\LWR@templengthone}{#3}%
        \LWR@BlockClassWP{%
            float:right; %
            width:\LWR@printlength{\LWR@templengthone};  % extra space
            margin:10pt%
        }{%
            width:\LWR@printlength{\LWR@templengthone}%
        }%
        (note)%
        {marginblock}%
    }%
    \renewcommand*{\@captype}{#2}%
}
{%
    \endLWR@BlockClassWP%
    \end{LWR@setvirtualpage}%
}
```

Env floatingfigure

[⟨*placement*⟩] {⟨*width*⟩}

```
\DeclareDocumentEnvironment{floatingfigure}{o m}
  {\begin{KFLTfloatflt@marginfloat}{figure}{#2}}
  {\end{KFLTfloatflt@marginfloat}}
```

Env floatingtable

[⟨*placement*⟩]

```
\DeclareDocumentEnvironment{floatingtable}{o}
```

```
{\begin{KFLTfloatflt@marginfloat}{table}{}}
{\end{KFLTfloatflt@marginfloat}}
```

File 169 **lwarp-floatpag.sty**

§ 278 Package floatpag

(*Emulates or patches code by* Vytas Statulevičius and Sigitas Tolušis.)

floatpag (*Pkg*) floatpag is ignored.

for HTML output: Discard all options for lwarp-floatpag:

```
\LWR@ProvidesPackageDrop{floatpag}[2012/05/29]

\newcommand*{\floatpagestyle}[1]{}
\newcommand*{\rotfloatpagestyle}[1]{}
\newcommand*{\thisfloatpagestyle}[1]{}
```

File 170 **lwarp-floatrow.sty**

§ 279 Package floatrow

(*Emulates or patches code by* Olga Lapko.)

floatrow (*Pkg*) floatrow is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{floatrow}[2008/08/02]
```

⚠ Misplaced alignment tab character &

Use \StartDefiningTabulars and \StopDefiningTabulars before and after defining macros using \ttabbox with a tabular inside. See section 8.10.1.

⚠ subfig package

When combined with the subfig package, while inside a subfloatrow \ffigbox and \ttabbox must have the caption in the first of the two of the mandatory arguments.

⚠ \FBwidth, \FBheight

The emulation of floatrow does not support \FBwidth or \FBheight. These values are pre-set to .3\linewidth and 2in. Possible solutions include:

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.
- Use warpprint and warpHTML environments to select appropriate values for each case.
- Inside a warpHTML environment, manually change \FBwidth or \FBheight before the \ffigbox or \ttabbox. Use \FBwidth or \FBheight normally afterwards; it will be used as expected in print output, and will use your custom-selected value in HTML output. This custom value will be used repeatedly, until it is manually changed to a new value.

After everything has loaded, remember whether subcaption was loaded. If not, it is assumed that subfig is used instead:

```
\newbool{LWR@subcaptionloaded}

\AtBeginDocument{
\IfPackageLoadedTF{subcaption}
    {\booltrue{LWR@subcaptionloaded}}
    {\boolfalse{LWR@subcaptionloaded}}
}
```

\floatbox [⟨*1 preamble*⟩] {⟨*2 captype*⟩} [⟨*3 width*⟩] [⟨*4 height*⟩] [⟨*5 vert pos*⟩] {⟨*6 caption*⟩} {⟨*7 object*⟩}

Only parameters for captype, width, caption, and object are used.

LWR@insubfloatrow is true if inside a subfloatrow environment.

There are two actions, depending on the use of subcaption or subfig.

```
\NewDocumentCommand{\floatbox}{o m o o o +m +m}{%
\ifbool{LWR@subcaptionloaded}%
{% subcaption
```

For subcaption:

```
    \ifbool{LWR@insubfloatrow}%
    {% subcaption in a subfloatrow
```

subfigure and subtable environments take width as an argument.

```
        \IfValueTF{#3}%
        {\@nameuse{sub#2}{#3}}%
        {\@nameuse{sub#2}{\linewidth}}%
    }% subcaption in a subfloatrow
    {% subcaption not in subfloatrow
```

figure and table environments do not take a width argument.

```
        \@nameuse{#2}%
    }% subcaption not in subfloatrow
    #6

    #7
```

End the environments:

```
    \ifbool{LWR@insubfloatrow}%
    {\@nameuse{endsub#2}}%
    {\@nameuse{end#2}}%
}% subcaption
{% assume subfig
```

For subfig:

```
\ifbool{LWR@insubfloatrow}%
{% subfig in a subfloatrow
```

\subfloat is a macro, not an environment.

Package subfig's \subfloat command takes an optional argument which is the caption, but \floatbox argument #6 contains commands to create the caption and label, not the caption itself. Thus, \caption is temporarily disabled to return its own argument without braces.

```
    \begingroup
    \let\caption\@firstofone
    \subfloat[#6]{#7}
    \endgroup
}% subfig in a subfloatrow
```

```
{% subfig package, but not a subfig
```

`figure` and `table` are environments:

```
\@nameuse{#2}
#6

#7
\@nameuse{end#2}
}% subfig package, but not a subfig
}% assume subfig
}
```

Not used:

```
\newcommand*{\nocapbeside}{}
\newcommand*{\capbeside}{}
\newcommand*{\captop}{}
\newlength{\FBwidth}
\setlength{\FBwidth}{.3\linewidth}
\newlength{\FBheight}
\setlength{\FBheight}{2in}
\newcommand*{\useFCwidth}{}
\newcommand{\floatsetup}[2][]{}
\newcommand{\thisfloatsetup}[1]{}
\newcommand{\clearfloatsetup}[1]{}
\newcommand*{\killfloatstyle}{}
```

`\newfloatcommand` {⟨*1 command*⟩} {⟨*2 captype*⟩} [⟨*3 preamble*⟩] [⟨*4 default width*⟩]

Preamble and default width are ignored.

```
\NewDocumentCommand{\newfloatcommand}{m m o o}{%
\@namedef{#1}{
\floatbox{#2}
}
}
```

`\renewfloatcommand` {⟨*1 command*⟩} {⟨*2 captype*⟩} [⟨*3 preamble*⟩] [⟨*4 default width*⟩]

Preamble and default width are ignored.

```
\NewDocumentCommand{\renewfloatcommand}{m m o o}{%
\@namedef{#1}{%
\floatbox{#2}
}
}
```

`\ffigbox` [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
\newfloatcommand{ffigbox}{figure}[\nocapbeside][]
```

`\ttabbox` [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
\newfloatcommand{ttabbox}{table}[\captop][\FBwidth]
```

`\fcapside` [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
\newfloatcommand{fcapside}{figure}[\capbeside][]
```

Env floatrow [⟨*numfloats*⟩]

The row of floats is placed into a of class floatrow.

```
\newenvironment*{floatrow}[1][2]
{%
    \begin{LWR@setvirtualpage}*%
    \BlockClass{floatrow}%
}
{
    \endBlockClass%
    \end{LWR@setvirtualpage}%
}
```

Keys for \DeclareNewFloatType:

```
\newcommand*{\LWR@frowkeyplacement}{}
\newcommand*{\LWR@frowkeyname}{}
\newcommand*{\LWR@frowkeyfileext}{}
\newcommand*{\LWR@frowkeywithin}{}
\newcommand*{\LWR@frowkeycapstyle}{}

\define@key{frowkeys}{placement}{}%
\define@key{frowkeys}{name}{\renewcommand{\LWR@frowkeyname}{#1}}%
\define@key{frowkeys}{fileext}{\renewcommand{\LWR@frowkeyfileext}{#1}}%
\define@key{frowkeys}{within}{\renewcommand{\LWR@frowkeywithin}{#1}}%
\define@key{frowkeys}{relatedcapstyle}{}%
```

\DeclareNewFloatType {⟨*type*⟩} {⟨*options*⟩}

Use \listof{type}{Title} to print a list of the floats.

```
\newcommand*{\DeclareNewFloatType}[2]{%
```

Reset key values:

```
\renewcommand*{\LWR@frowkeyplacement}{}%
\renewcommand*{\LWR@frowkeyname}{}%
\renewcommand*{\LWR@frowkeyfileext}{}%
\renewcommand*{\LWR@frowkeywithin}{}%
\renewcommand*{\LWR@frowkeycapstyle}{}%
```

Read new key values:

```
\LWR@traceinfo{about to setkeys frowkeys}%
\setkeys{frowkeys}{#2}%
\LWR@traceinfo{finished setkeys frowkeys}%
```

Create a new float with optional [within]:

```
\ifthenelse{\equal{\LWR@frowkeywithin}{}}%
{%
    \DeclareFloatingEnvironment[
        placement=\LWR@frowkeyplacement,
        fileext=\LWR@frowkeyfileext
    ]{#1}%
}%
{%
    \DeclareFloatingEnvironment[
        placement=\LWR@frowkeyplacement,
        fileext=\LWR@frowkeyfileext,
        within=\LWR@frowkeywithin
    ]{#1}%
%     \LWR@traceinfo{finished newfloat #1}%
```

```
}%
```

Rename the float if a name was given:

```
\ifthenelse{\equal{\LWR@frowkeyname}{}}%
    {}%
    {%
        \SetupFloatingEnvironment{#1}{name={\LWR@frowkeyname}}%
    }%
}
```

Not used:

```
\newcommand{\buildFBBOX}[2]{}
\newcommand*{\CenterFloatBoxes}{}
\newcommand*{\TopFloatBoxes}{}
\newcommand*{\BottomFloatBoxes}{}
\newcommand*{\PlainFloatBoxes}{}

\newcommand{\capsubrowsettings}{}

\NewDocumentCommand{\RawFloats}{o o}{}
```

`\RawCaption` {⟨*text*⟩}

To be used inside a minipage or parbox.

```
\newcommand{\RawCaption}[1]{#1}
```

`\floatfoot` {⟨*text*⟩}

Places additional text inside a float, inside a css `<div>` of class `floatfoot`.

```
\NewDocumentCommand{\floatfoot}{s +m}{%
    \begin{BlockClass}{floatfoot}
    #2
    \end{BlockClass}
}
```

Used to compute `\linewidth`.

```
\newbool{LWR@insubfloatrow}
\boolfalse{LWR@insubfloatrow}
```

Env `subfloatrow` [⟨*num_floats*⟩]

```
\newenvironment*{subfloatrow}[1][2]
{
```

The row of floats is placed into a `<div>` of class `floatrow`:

```
    \LWR@forcenewpage
    \BlockClass{floatrow}
```

While inside the floatrow, `LWR@insubfloatrow` is set true, which tells `\floatbox` to use `\subfigure` or `\subtable`.

```
    \begingroup%
    \booltrue{LWR@insubfloatrow}%
}
{%
    \endgroup%
```

```
    \endBlockClass%
    \boolfalse{LWR@insubfloatrow}%
}
```

File 171 **lwarp-fltrace.sty**

§ 280 Package

fltrace

`fltrace` (*Pkg*) fltrace is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fltrace}[2018/01/08]

\def\tracefloats{}
\def\tracefloatsoff{}
\def\tracefloatvals{}
```

File 172 **lwarp-flushend.sty**

§ 281 Package

flushend

(*Emulates or patches code by* Sigitas Tolušis.)

`flushend` (*Pkg*) flushend is ignored.

for HTML output: Discard all options for lwarp-flushend:

```
\LWR@ProvidesPackageDrop{flushend}[2021/10/04]

\newcommand*{\flushend}{}
\newcommand*{\raggedend}{}
\newcommand*{\flushcolsend}{}
\newcommand*{\raggedcolsend}{}
\newtoks\atColsBreak \atColsBreak={}
\newtoks\atColsEnd   \atColsEnd={}
\newcommand*{\showcolsendrule}{}
```

File 173 **lwarp-fnbreak.sty**

§ 282 Package

fnbreak

`fnbreak` (*Pkg*) fnbreak is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnbreak}[2012/01/01]

\newcommand*{\fnbreakverbose}{}
\newcommand*{\fnbreaknonverbose}{}
\newcommand*{\fnbreaklabel}{}
\newcommand*{\fnbreaknolabel}{}
```

File 174 **lwarp-fncychap.sty**

§ 283 Package fncychap

(*Emulates or patches code by* Ulf A. Lindgren.)

fncychap (*Pkg*) fncychap is ignored.

for HTML output: Discard all options for lwarp-fncychap:

```
\LWR@ProvidesPackageDrop{fncychap}[2007/07/30]

\def\mghrulefill#1{}
\def\ChNameLowerCase{}
\def\ChNameUpperCase{}
\def\ChNameAsIs{}
\def\ChTitleLowerCase{}
\def\ChTitleUpperCase{}
\def\ChTitleAsIs{}
\newcommand{\ChRuleWidth}[1]{}
\newcommand{\ChNameVar}[1]{}
\newcommand{\ChNumVar}[1]{}
\newcommand{\ChTitleVar}[1]{}
\newcommand{\TheAlphaChapter}{}
\newcommand{\DOCH}{}
\newcommand{\DOTI}[1]{}
\newcommand{\DOTIS}[1]{}
\newlength{\mylen}
\newlength{\myhi}
\newlength{\px}
\newlength{\py}
\newlength{\pyy}
\newlength{\pxx}
\newlength{\RW}
\newcommand{\FmN}[1]{#1}
\newcommand{\FmTi}[1]{#1}
```

File 175 **lwarp-fnlineno.sty**

§ 284 Package fnlineno

fnlineno (*Pkg*) fnlineno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnlineno}[2011/01/07]
```

File 176 **lwarp-fnpara.sty**

§ 285 Package fnpara

fnpara (*Pkg*) fnpara is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnpara}
```

File 177 **lwarp-fnpos.sty**

§ 286 Package

fnpos

(*Emulates or patches code by* Hiroshi Nakashima.)

fnpos (*Pkg*) fnpos is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnpos}[1999/07/14]

\newcommand*{\makeFNbottom}{}
\newcommand*{\makeFNmid}{}
\newcommand*{\makeFNbelow}{}
\newcommand*{\makeFNabove}{}
```

File 178 **lwarp-fontawesome.sty**

§ 287 Package

fontawesome

(*Emulates or patches code by* Xavier Danaux.)

fontawesome (*Pkg*) fontawesome is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

⚠ *poppler* syntax warning If using pdf LaTeX, *poppler* may issue a syntax warning regarding parsing a ligature component. XeLaTeX or LuaLaTeX may be used to avoid this warning.

In the following, the general strategy is to intercept `\symbol` and embed it inside a `lateximage`. These changes are done inside a local group.

For pdf LaTeX, the `alt` tag includes the icon (symbol) number. For XeLaTeX and LuaLaTeX, the `alt` tag is generic.

for HTML output:

```
\LWR@ProvidesPackagePass{fontawesome}[2016/05/15]

\LetLtxMacro\LWR@orig@symbol\symbol

\ifxetexorluatex

\newfontfamily{\LWR@orig@FA}{FontAwesome}

\newcommand*{\LWR@fontawesome@xelatex@symbol}[1]{%
    \LWR@findcurrenttextcolor%
   \begin{lateximage}*[icon][fontawesomexetex#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
    \csuse{\LWR@font@size}%
    \LWR@orig@FA%
    \LWR@orig@symbol{#1}%
    \end{lateximage}%
}

\RenewDocumentCommand{\FA}{}{%
```

```
    \LetLtxMacro\symbol\LWR@fontawesome@xelatex@symbol%
}

\else

\newcommand*{\LWR@fontawesome@symbolX}[2]{%
    \LWR@findcurrenttextcolor%
   \begin{lateximage}*[icon #1][fontawesome#2#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
    \csuse{\LWR@font@size}%
    \fontencoding{U}\fontfamily{fontawesome#2}\selectfont%
    \LWR@orig@symbol{#1}%
    \end{lateximage}%
}

\newcommand*{\LWR@fontawesome@symbolone}[1]{%
    \LWR@fontawesome@symbolX{#1}{one}%
}

\newcommand*{\LWR@fontawesome@symboltwo}[1]{%
    \LWR@fontawesome@symbolX{#1}{two}%
}

\newcommand*{\LWR@fontawesome@symbolthree}[1]{%
    \LWR@fontawesome@symbolX{#1}{three}%
}

\renewrobustcmd\FAone{%
    \LetLtxMacro\symbol\LWR@fontawesome@symbolone%
}

\renewrobustcmd\FAtwo{%
    \LetLtxMacro\symbol\LWR@fontawesome@symboltwo%
}

\renewrobustcmd\FAthree{%
    \LetLtxMacro\symbol\LWR@fontawesome@symbolthree%
}
\fi
```

File 179 **lwarp-fontawesome5.sty**

§ 288 Package

fontawesome5

(*Emulates or patches code by* Marcel Krüger.)

`fontawesome5` (*Pkg*) fontawesome5 is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

The `alt` tag has the name of the icon.

for HTML output:

```
\LWR@ProvidesPackagePass{fontawesome5}[2018/07/27]

\ExplSyntaxOn
\cs_set:Nn\fontawesome_use_icon:nn{
    \LWR@findcurrenttextcolor
  \cs_if_exist:cTF{c__fontawesome_slot_#2_tl}{
```

```
  \begin{lateximage}*[#2][fontawesome5#1SZ\LWR@font@size{}CL\LWR@tempcolor]
    \csuse{\LWR@font@size}
    \exp_last_unbraced:Nv
      \__fontawesome_icon_at:nnnn
      {c__fontawesome_slot_#2_tl}
        {#1}{#2}
    \end{lateximage}
  }{
    \msg_error:nnxx{fontawesome5}{icon-not-found}{#2}{#1}
  }
}
\ExplSyntaxOff
```

File 180 **lwarp-fontaxes.sty**

§ 289 Package fontaxes

(*Emulates or patches code by* Andreas Bühmann, Michael Ummels.)

fontaxes (*Pkg*) fontaxes is emulated for HTML, and used as-is for print output.

Functionality for small caps is in the lwarp core. Swashes and figure styles are ignored for HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{fontaxes}[2014/03/23]

\ifdef{\LWR@HTML@swshape}{}{% duplicated by nfssext-cfr
    \newcommand{\LWR@HTML@swshape}{}
    \LWR@formatted{swshape}

    \newrobustcmd{\LWR@HTML@textsw}[1]{#1}
    \LWR@formatted{textsw}

    \FilenameNullify{%
        \LetLtxMacro\swshape\@empty%
        \LetLtxMacro\textsw\firstofone%
    }
}
```

File 181 **lwarp-fontenc.sty**

§ 290 Package fontenc

fontenc (*Pkg*) If using pdf LaTeX, lwarp used to require fontenc be loaded before lwarp, but now lwarp itself loads \fontenc with T1 encoding, which lwarp requires. fontenc is now allowed to be loaded with another encoding after lwarp.

lwarp-fontenc is no longer necessary, but is still provided to overwrite older versions.

for HTML output:

```
\LWR@ProvidesPackagePass{fontenc}[2017/04/05]
```

File 182 **lwarp-footmisc.sty**

§ 291 Package

footmisc

(*Emulates or patches code by* Robin Fairbairns.)

`footmisc` (*Pkg*) footmisc is emulated.

lwarp incidentally happens to emulate the `stable` option.

```
\LWR@ProvidesPackageDrop{footmisc}[2011/06/06]
```

Some nullified commands:

```
\newcommand{\footnotelayout}{}
\newcommand{\setfnsymbol}[1]{}
\NewDocumentCommand{\DefineFNsymbols}{s m o m}{}

\newdimen\footnotemargin
\footnotemargin1.8em\relax

\newcommand*\hangfootparskip{0.5\baselineskip}
\newcommand*\hangfootparindent{0em}%

\let\pagefootnoterule\footnoterule
\let\mpfootnoterule\footnoterule
\def\splitfootnoterule{\kern-3\p@ \hrule \kern2.6\p@}

\providecommand*{\multiplefootnotemarker}{3sp}
\providecommand*{\multfootsep}{,}
```

Using cleveref. `\labelcref` only prints the number of the object, not its type.

```
\providecommand*{\footref}[1]{\labelcref{#1}}
```

The following work as-is:

```
\newcommand\mpfootnotemark{%
  \@ifnextchar[%
    \@xmpfootnotemark%
    {%
      \stepcounter\@mpfn%
      \protected@xdef\@thefnmark{\thempfn}%
      \@footnotemark%
    }%
}
\def\@xmpfootnotemark[#1]{%
  \begingroup%
    \csname c@\@mpfn\endcsname #1\relax%
    \unrestored@protected@xdef\@thefnmark{\thempfn}%
  \endgroup%
  \@footnotemark%
}
```

File 183 **lwarp-footnote.sty**

§ 292 Package

footnote

(*Emulates or patches code by* Mark Wooding.)

footnote (*Pkg*) footnote is used with minor patches.

for HTML output: footnote patches \@makefntext in a strange way. It must be restored to the expected defintion before loading footnote, then replaced again after.

```
\long\def\@makefntext#1{\textsuperscript{\@thefnmark}~#1}

\LWR@ProvidesPackagePass{footnote}[1997/01/28]

\long\def\@makefntext#1{\textsuperscript{\@thefnmark}~{#1}}


\def\spewnotes{%
  \endgroup%
  \if@savingnotes\else\ifvoid\fn@notes\else\begingroup%
    \let\@makefntext\@empty%
    \let\@finalstrut\@gobble%
    \let\rule\@gobbletwo%
    \booltrue{LWR@spewingnotes}%        lwarp
    \@footnotetext{\unvbox\fn@notes}%
  \endgroup\fi\fi%
}
\let\endsavenotes\spewnotes


\def\fn@fntext#1{%
  \ifx\ifmeasuring@\@@undefined%
    \expandafter\@secondoftwo\else\expandafter\@iden%
  \fi%
  {\ifmeasuring@\expandafter\@gobble\else\expandafter\@iden\fi}%
  {%
    \global\setbox\fn@notes\vbox{%
      \unvbox\fn@notes%
      \LWR@htmltagc{\LWR@tagregularparagraph}%      lwarp
      \LWR@orignewline%                             lwarp
      \fn@startnote%
      \@makefntext{%
        \rule\z@\footnotesep%
        \ignorespaces%
        #1%
        \@finalstrut\strutbox%
      }%
      \fn@endnote%
    }%
  }%
}
```

Removed print-version formatting:

```
\def\fn@startnote{%
```

```
%   \@parboxrestore%
  \protected@edef\@currentlabel{\csname p@\@mpfn\endcsname\@thefnmark}%
%   \color@begingroup% *** conflicts with lwarp
}

% \let\fn@endnote\color@endgroup% *** conflicts with lwarp
\def\fn@endnote{%
    \LWR@origtilde\LWR@orignewline%
    \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
    \LWR@origtilde\LWR@orignewline%
}
```

Removed print-version formatting:

```
\def\fn@startfntext{%
  \setbox\z@\vbox\bgroup%
    \LWR@htmltagc{\LWR@tagregularparagraph}%    lwarp
    \LWR@orignewline%                           lwarp
    \fn@startnote%
    \fn@prefntext% Req'd for numbering.
%     \rule\z@\footnotesep%
    \ignorespaces%
}

```

Removed print-version formatting, added closing paragraph tag:

```
\def\fn@endfntext{%
    \fn@postfntext%
    \LWR@origtilde\LWR@orignewline%
    \LWR@htmltagc{/\LWR@tagregularparagraph}%
    \LWR@orignewline%
  \egroup%
  \begingroup%
    \let\@makefntext\@empty%
    \let\@finalstrut\@gobble%

    \LetLtxMacro\rule\@gobbletwo%
    \booltrue{LWR@spewingnotes}%    lwarp
    \@footnotetext{\unvbox\z@}%
  \endgroup%
}
```

These have been redefined, so re-`\let` them again:

```
\let\endfootnote\fn@endfntext
\let\endfootnotetext\endfootnote
```

File 184 **lwarp-footnotebackref.sty**

§ 293 Package

footnotebackref

`footnotebackref` (*Pkg*) footnotebackref is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{footnotebackref}[2012/07/01]
```

File 185 **lwarp-footnotehyper.sty**

§ 294 Package
# footnotehyper

footnotehyper (*Pkg*) footnotehyper is a hyperref-safe version of footnote. For lwarp, footnotehyper is emulated.

for HTML output: Discard all options for lwarp-footnotehyper:

```
\RequirePackage{footnote}

\LWR@ProvidesPackageDrop{footnotehyper}[2018/01/23]
```

File 186 **lwarp-footnoterange.sty**

§ 295 Package
# footnoterange

(*Emulates or patches code by* H.-Martin Münch.)

footnoterange (*Pkg*) footnoterange is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{footnoterange}[2012/02/17]

\csletcs{footnoterange}{footnoterange*}
\csletcs{endfootnoterange}{endfootnoterange*}
```

File 187 **lwarp-footnpag.sty**

§ 296 Package
# footnpag

footnpag (*Pkg*) footnpag is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{footnpag}
```

File 188 **lwarp-foreign.sty**

§ 297 Package
# foreign

(*Emulates or patches code by* Philip G. Ratcliffe.)

foreign (*Pkg*) foreign is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{foreign}[2012/09/25]

\renewcommand\foreignabbrfont{\emph}
```

File 189 **lwarp-forest.sty**

§ 298 Package forest

(*Emulates or patches code by* Sašo Živanović.)

forest (*Pkg*) forest is patched for use by lwarp.

⚠ \Forest* The starred version of the macro \Forest* is not supported. lwarp encases each lateximage in an environment, so the global results of the starred \Forest* are lost.

for HTML output:

```
\LWR@ProvidesPackagePass{forest}[2017/07/14]

\BeforeBeginEnvironment{forest}{%
    \begin{lateximage}[-forest-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{forest}{\end{lateximage}}

\RenewDocumentCommand{\Forest}{s D(){} m}{%
  \forest@config{#2}%
  \IfBooleanTF{#1}{%
        \PackageError{lwarp-forest}%
        {\protect\Forest* is not supported}%
        {Lwarp uses an environment for images,\MessageBreak
            but \protect\Forest* cannot work in an environment.}%
        \let\forest@next\forest@env%
    }{\let\forest@next\forest@group@env}%
    \begin{lateximage}[-forest-~\PackageDiagramAltText]%     lwarp
  \forest@next{#3}%
    \end{lateximage}%               lwarp
}
```

File 190 **lwarp-fouridx.sty**

§ 299 Package fouridx

(*Emulates or patches code by* Stefan Karrmann.)

fouridx (*Pkg*) fouridx works as-is with SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{fouridx}[2013/11/21]

\begin{warpMathJax}
\CustomizeMathJax{%
    \newcommand{\fourIdx}[5]{%
        \vphantom{#5}^{\hphantom{#2}#1}_{\hphantom{#1}#2}{#5}^{#3}_{#4}%
    }%
}
\end{warpMathJax}
```

File 191 **lwarp-fourier.sty**

§ 300 Package

fourier

(*Emulates or patches code by* Michel Bovani.)

`fourier` (*Pkg*) fourier is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options, except `sloped` and `upright` are honored for Greek characters, but MathJax cannot yet honor these for Latin characters.

The dedicated macros for upright and italic Greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{fourier}[2020/03/03]

\LWR@infoprocessingmathjax{fourier}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}

\begin{warpMathJax}

\IfPackageLoadedWithOptionsTF{fourier}{sloped}
    {
        \LWR@mathjax@addgreek@l@up{other}{}
        \LWR@mathjax@addgreek@u@it*{other}{}
    }% sloped
    {% not sloped
        \IfPackageLoadedWithOptionsTF{fourier}{upright}
            {% upright option
                \LWR@mathjax@addgreek@l@up{}{}
                \LWR@mathjax@addgreek@u@up*{}{}
                \LWR@mathjax@addgreek@l@it{other}{}
                \LWR@mathjax@addgreek@u@it*{other}{}
            }
            {% neither sloped nor upright
                \LWR@mathjax@addgreek@l@up{other}{}
                \LWR@mathjax@addgreek@u@it*{other}{}
            }
    }

\CustomizeMathJax{\newcommand{\othergreek}[1]{#1}}
\CustomizeMathJax{\let\varvarrho\varrho}
\CustomizeMathJax{\let\varvarpi\varpi}
\CustomizeMathJax{\let\othervarvarpi\othervarpi}
\CustomizeMathJax{\let\othervarvarrho\othervarrho}
\CustomizeMathJax{\let\varpartialdiff\partial}
```

`lwarp_mathjax.txt` adds `\left`/`\right` support for delimiters.

```
\CustomizeMathJax{\let\llbracket\lBrack}
```

```
\CustomizeMathJax{\let\rrbracket\rBrack}
\CustomizeMathJax{\let\dblbrackleft\lBrack}
\CustomizeMathJax{\let\dblbrackright\rBrack}

\CustomizeMathJax{\let\VERT|}

\CustomizeMathJax{\newcommand{\parallelslant}{\mathrel{\unicode{x02AFD}}}}
\CustomizeMathJax{\newcommand{\thething}{\mathord{\unicode{x1F60E}}}}
\CustomizeMathJax{\newcommand{\nparallelslant}{%
    \mathrel{\LWRoverlaysymbols{-}{\unicode{x02AFD}}}%
}}
\CustomizeMathJax{\newcommand{\xswordsup}{\mathord{\unicode{x2694}}}}
\CustomizeMathJax{\newcommand{\xswordsdown}{\mathord{\unicode{x2694}}}}% up
\CustomizeMathJax{\newcommand{\notowns}{\mathrel{\unicode{x220C}}}}

\CustomizeMathJax{\newcommand{\iintop}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiintop}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\let\oiintop\oiint}
\CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
\CustomizeMathJax{\let\oiiintop\oiiint}
\CustomizeMathJax{\newcommand{\slashint}{\mathop{\unicode{x2A0D}}\limits}}
\CustomizeMathJax{\let\slashintop\slashint}

\CustomizeMathJax{\let\overgroup\overparen}
\CustomizeMathJax{\let\wideparen\overparen}
\CustomizeMathJax{\let\widearc\overparen}
\CustomizeMathJax{\let\wideOarc\overrightarrow}
\CustomizeMathJax{\newcommand{\widering}[1]{\stackrel{\unicode{x2218}}{\overgroup{#1}}}}

\end{warpMathJax}
```

File 192 **lwarp-framed.sty**

§ 301 Package framed

(*Emulates or patches code by* Donald Arseneau.)

framed (*Pkg*) framed is supported and patched by lwarp.

for HTML output: Accept all options for lwarp-framed:

```
\LWR@ProvidesPackagePass{framed}[2011/10/22]

\AtBeginDocument{\RequirePackage{xcolor}}% for \convertcolorspec

\renewenvironment{framed}
{%
    \LWR@forcenewpage
    \BlockClass{framed}%
}
{\endBlockClass}

\renewenvironment{oframed}
{%
    \LWR@forcenewpage
    \BlockClass{framed}%
}
```

```
{\endBlockClass}


\renewenvironment{shaded}
{%
    \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
    \LWR@forcenewpage
    \BlockClass[background: \LWR@origpound\LWR@tempcolor]{shaded}%
}
{\endBlockClass}

\renewenvironment{shaded*}
{%
    \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
    \LWR@forcenewpage
    \BlockClass[background: \LWR@origpound\LWR@tempcolor]{shaded}%
}
{\endBlockClass}


\renewenvironment{leftbar}{%
    \LWR@forcenewpage
    \BlockClass{framedleftbar}
    \def\FrameCommand{}%
    \MakeFramed {}
}%
{\endMakeFramed\endBlockClass}


\renewenvironment{snugshade}
{%
    \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
    \LWR@forcenewpage
    \BlockClass[background: \LWR@origpound\LWR@tempcolor]{snugframed}%
}
{\endBlockClass}

\renewenvironment{snugshade*}
{%
    \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
    \LWR@forcenewpage
    \BlockClass[background: \LWR@origpound\LWR@tempcolor]{snugframed}%
}
{\endBlockClass}

\let\oframed\framed
\let\endoframed\endframed


\RenewEnviron{titled-frame}[1]{%
    \CustomFBox{#1}{}{0pt}{0pt}{0pt}{0pt}{\BODY}
}
```

\CustomFBox {⟨*toptitle*⟩} {⟨*bottitle*⟩} {⟨*thicknesstop*⟩} {⟨*bottom*⟩} {⟨*left*⟩} {⟨*right*⟩} {⟨*text contents*⟩}

```
\renewcommand{\CustomFBox}[7]{%
    \convertcolorspec{named}{TFFrameColor}{HTML}\LWR@tempcolor%
    \LWR@forcenewpage
```

```
\begin{BlockClass}[border: 3px solid \LWR@origpound\LWR@tempcolor]{framed}%
 \ifthenelse{\isempty{#1}}{}{% not empty
  \begin{BlockClass}[background: \LWR@origpound\LWR@tempcolor]{framedtitle}%
     \textcolor{TFTitleColor}{\textbf{#1}}%
     \end{BlockClass}%
 }% not empty

 #7

 \ifthenelse{\isempty{#2}}{}{% not empty
     \convertcolorspec{named}{TFFrameColor}{HTML}\LWR@tempcolor%
  \begin{BlockClass}[background: \LWR@origpound\LWR@tempcolor]{framedtitle}%
     \textcolor{TFTitleColor}{\textbf{#2}}%
     \end{BlockClass}%
 }% not empty
 \end{BlockClass}%
}
```

\TitleBarFrame [⟨*marker*⟩] {⟨*title*⟩} {⟨*contents*⟩}

```
\renewcommand\TitleBarFrame[3][]{%
    \CustomFBox%
        {#2}{}%
        \fboxrule\fboxrule\fboxrule\fboxrule%
        {#3}%
}

\renewcommand{\TF@Title}[1]{#1}
```

Env MakeFramed {⟨*settings*⟩}

```
\let\MakeFramed\relax
\let\endMakeFramed\relax

\NewEnviron{MakeFramed}[1]{%
    \FrameCommand{\begin{minipage}{\linewidth}\BODY\end{minipage}}%
}
```

\fb@put@frame {⟨*frame cmd no split*⟩} {⟨*frame cmd split*⟩}

```
\renewcommand*{\fb@put@frame}[2]{%
    \relax%
    \@tempboxa%
}
```

File 193 **lwarp-froufrou.sty**

§ 302 Package **froufrou**

(*Emulates or patches code by* Nelson Lago.)

froufrou (*Pkg*) froufrou is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{froufrou}[2020/12/22]

\ExplSyntaxOn
```

```
\xpretocmd{\setfroufrou}
    {\edef\LWR@latestfroufrou{\detokenize{#1}}}
    {}
    {\LWR@patcherror{froufrou}{setfroufrou}}
\ExplSyntaxOff

\RenewDocumentCommand{\froufrou}{s O{}}{%
  \nopagebreak[4]\par

  \IfBooleanTF{#1}{\@afterindenttrue}{\@afterindentfalse}

  \nopagebreak[4]\@froufrouspacebefore\nopagebreak[4]

  \bgroup
    \setfroufrou{#2}%
    \normalsize
   \ifdefvoid{\setstretch}{}{\setstretch{\setspace@singlespace}}% normally 1
    \setlength{\parskip}{0pt}
    \noindent\centering\bgroup%
        \begin{center}%                                          lwarp
        \begin{lateximage}*[froufrou][\LWR@latestfroufrou]%      lwarp
        \@froufrouOrnament%
        \end{lateximage}%                                        lwarp
        \end{center}%                                            lwarp
    \egroup\par
  \egroup

  \nopagebreak[4]\@froufrouspaceafter\nopagebreak[4]

  \@froufrouFixSpacingAfter

  \nopagebreak[3]

  \@afterheading
}
```

File 194 **lwarp-ftcap.sty**

§ 303 Package

ftcap

`ftcap` (*Pkg*) ftcap is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{ftcap}
```

File 195 **lwarp-ftnright.sty**

§ 304 Package

ftnright

`ftnright` (*Pkg*) ftnright is ignored.

for HTML output: Discard all options for lwarp-ftnright:

```
\LWR@ProvidesPackageDrop{ftnright}[2014/10/28]
```

File 196 **lwarp-fullminipage.sty**

§ 305 Package fullminipage

fullminipage (*Pkg*) fullminipage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fullminipage}[2014/07/06]
\newenvironment{fullminipage}[1][]{}{}
```

File 197 **lwarp-fullpage.sty**

§ 306 Package fullpage

fullpage (*Pkg*) fullpage is ignored.

for HTML output: Discard all options for lwarp-fullpage:

```
\LWR@ProvidesPackageDrop{fullpage}[1994/06/01]
```

File 198 **lwarp-fullwidth.sty**

§ 307 Package fullwidth

(*Emulates or patches code by* Marco Daniel.)

fullwidth (*Pkg*) fullwidth is emulated.

A minipage is used, of no HTML width.

for HTML output:

```
\LWR@ProvidesPackageDrop{fullwidth}[2011/11/18]
\newenvironment*{fullwidth}[1][]{%
\minipagefullwidth%
\minipage{\linewidth}%
}
{%
\endminipage%
}
```

File 199 **lwarp-fvextra.sty**

§ 308 Package fvextra

(*Emulates or patches code by* Geoffrey M. Poore.)

fvextra (*Pkg*) fvextra is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{fvextra}[2022/11/30]
```

Ignored are highlight, showtabs, obeytabs, tab, tabcolor. Also ignored are all options regarding line breaking except breaklines, which is emulated as true.

tabsize is honored.

If line numbers on the right side are used along with breaklines, the line numbers will not be aligned.

```
\define@booleankey{FV}{obeytabs}%
%   {\let\FV@ObeyTabsInit\FV@@ObeyTabsInit}%
  {\let\FV@ObeyTabsInit\relax}
  {\let\FV@ObeyTabsInit\relax}

\define@key{FV}{tabcolor}%
 {}

\define@key{FV}{tab}{}

\define@booleankey{FV}{showtabs}%
%  {\def\FV@TabChar{\FV@TabColor{\FancyVerbTab}}}%
 {\let\FV@TabChar\relax}
 {\let\FV@TabChar\relax}

\newbool{LWR@FV@breaklines}

\define@booleankey{FV}{breaklines}%
  {\boolfalse{FV@breaklines}
    \booltrue{LWR@FV@breaklines}
    \let\FV@ListProcessLine\FV@ListProcessLine@NoBreak}
  {\boolfalse{FV@breaklines}
    \boolfalse{LWR@FV@breaklines}
    \let\FV@ListProcessLine\FV@ListProcessLine@NoBreak}
% \fvset{breaklines}

\define@key{FV}{breakanywheresymbolpre}{\def\FancyVerbBreakAnywhereSymbolPre{}}
\fvset{breakanywheresymbolpre={}}

\define@key{FV}{breakanywheresymbolpost}{\def\FancyVerbBreakAnywhereSymbolPost{}}
\fvset{breakanywheresymbolpost={}}

\define@key{FV}{breakbeforesymbolpre}{\def\FancyVerbBreakBeforeSymbolPre{}}
\fvset{breakbeforesymbolpre={}}

\define@key{FV}{breakbeforesymbolpost}{\def\FancyVerbBreakBeforeSymbolPost{}}
\fvset{breakbeforesymbolpost={}}

\define@key{FV}{breakaftersymbolpre}{\def\FancyVerbBreakAfterSymbolPre{}}
\fvset{breakaftersymbolpre={}}

\define@key{FV}{breakaftersymbolpost}{\def\FancyVerbBreakAfterSymbolPost{}}
\fvset{breakaftersymbolpost={}}

\define@key{FV}{breaksymbolleft}{\def\FancyVerbBreakSymbolLeft{}}

\define@key{FV}{breaksymbol}{\fvset{breaksymbolleft={}}}

\fvset{breaksymbolleft={}}

\define@key{FV}{breaksymbolright}{\def\FancyVerbBreakSymbolRight{}}
\fvset{breaksymbolright={}}
```

```
54
55 \def\FV@ListProcessLine@NoBreak#1{%
56 %   \hbox to \hsize{%
57 %     \kern\leftmargin
58 %     \hbox to \linewidth{%
59       \FV@LeftListNumber%
60       \FV@LeftListFrame%
61       \FancyVerbFormatLine{%
62         \FancyVerbHighlightLine{%
63           \FV@ObeyTabs{\FancyVerbFormatText{#1}}}}%\hss
64       \FV@RightListFrame%
65       \FV@RightListNumber%
66 %     }%
67 %     \hss}%
68 \null\par%                    lwarp
69 }
70
71
72 \newcommand*{\LWR@FV@linethensep}{%
73     \ifbool{LWR@FV@breaklines}%
74         {\theFancyVerbLine\kern\FV@NumberSep}%
75         {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}%
76 }
77
78 \newcommand*{\LWR@FV@septhenline}{%
79     \ifbool{LWR@FV@breaklines}%
80         {\kern\FV@NumberSep\theFancyVerbLine}%
81         {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}%
82 }
83
84 \xpatchcmd{\FV@Numbers@left}
85     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
86     {\LWR@FV@linethensep}
87     {}
88     {\LWR@patcherror{fvextra}{FV@Numbers@left A}}
89
90 \xpatchcmd{\FV@Numbers@left}
91     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
92     {\LWR@FV@linethensep}
93     {}
94     {\LWR@patcherror{fvextra}{FV@Numbers@left B}}
95
96 \xpatchcmd{\FV@Numbers@left}
97     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
98     {\LWR@FV@linethensep}
99     {}
100    {\LWR@patcherror{fvextra}{FV@Numbers@left C}}
101
102 \xpatchcmd{\FV@Numbers@right}
103    {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
104    {\LWR@FV@septhenline}
105    {}
106    {\LWR@patcherror{fvextra}{FV@Numbers@right A}}
107
108 \xpatchcmd{\FV@Numbers@right}
109    {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
110    {\LWR@FV@septhenline}
111    {}
112    {\LWR@patcherror{fvextra}{FV@Numbers@right B}}
113
```

```
\xpatchcmd{\FV@Numbers@right}
    {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
    {\LWR@FV@linethensep}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@right C}}

\xpatchcmd{\FV@Numbers@both}
    {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
    {\LWR@FV@linethensep}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@both A}}

\xpatchcmd{\FV@Numbers@both}
    {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
    {\LWR@FV@linethensep}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@both B}}

\xpatchcmd{\FV@Numbers@both}
    {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
    {\LWR@FV@linethensep}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@both C}}

\xpatchcmd{\FV@Numbers@both}
    {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
    {\LWR@FV@septhenline}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@both D}}

\xpatchcmd{\FV@Numbers@both}
    {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
    {\LWR@FV@septhenline}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@both E}}

\xpatchcmd{\FV@Numbers@both}
    {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
    {\LWR@FV@linethensep}
    {}
    {\LWR@patcherror{fvextra}{FV@Numbers@both F}}
```

File 200 **lwarp-fwlw.sty**

§ 309 Package

fwlw

`fwlw` (*Pkg*) fwlw is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fwlw}

\newbox\FirstWordBox      \global\setbox\FirstWordBox\hbox{}
\newbox\NextWordBox       \global\setbox\NextWordBox\hbox{}
\newbox\LastWordBox       \global\setbox\LastWordBox\hbox{}
\def\ps@fwlwhead{}
\def\ps@NextWordFoot{}
```

File 201 **lwarp-gensymb.sty**

§ 310 Package gensymb

(*Emulates or patches code by* Walter Schmidt.)

gensymb (*Pkg*) gensymb works as-is for svg math, and uses the MathJax package.

for HTML output:

```
\LWR@ProvidesPackagePass{gensymb}[2003/07/02]

\begin{warpMathJax}
\CustomizeMathJax{\require{gensymb}}
\end{warpMathJax}
```

File 202 **lwarp-gentombow.sty**

§ 311 Package gentombow

gentombow (*Pkg*) gentombow is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{gentombow}[2018/05/17]

\newcommand{\settombowbanner}[1]{}
\newcommand{\settombowbannerfont}[1]{}
\newcommand{\settombowwidth}[1]{}
\newcommand{\settombowbleed}[1]{}
\newcommand{\settombowcolor}[1]{}
```

File 203 **lwarp-geometry.sty**

§ 312 Package geometry

(*Emulates or patches code by* Hideo Umeki.)

geometry (*Pkg*) geometry is preloaded by lwarp, but must be nullified as seen by the user's source code.

for HTML output: Discard all options for lwarp-geometry:

```
\LWR@ProvidesPackageDropA{geometry}{2018/04/16}
```

If geometry is never loaded by the user, it will be loaded by lwarp `\AtBeginDocument`. If this is the case, the page layout should not be changed but the user macros should still be nullified.

```
\ifbool{LWR@allowanothergeometry}{%
```

Assign and set the selected geometry with `reset` prepended. `\AtEndPreamble` lwarp will save this, then set its own geometry.

```
    \edef\LWR@tempone{reset,\@ptionlist{\@currname.\@currext}}%
    \expandafter\LWR@origgeometry\expandafter{\LWR@tempone}%
}{}% LWR@allowanothergeometry
```

The user-level commands are nullified:

```
\renewcommand*{\geometry}[1]{}
\renewcommand*{\newgeometry}[1]{}
\renewcommand*{\restoregeometry}{}
\renewcommand*{\savegeometry}[1]{}
\renewcommand*{\loadgeometry}[1]{}
```

File 204 **lwarp-ghsystem.sty**

§ 313 Package ghsystem

(*Emulates or patches code by* Clemens Niederberger.)

`ghsystem` (*Pkg*) ghsystem is patched for use by lwarp.

⚠ `\ghspic` images Images must be provided in SVG format, unless JPG is specified. It is recommended to create a local `images` directory, copy into it the relevent PDF ghsystem images, and then convert them with

Enter ⇒ **`lwarpmk pdftosvg images/*.pdf`**

for HTML output:

```
\LWR@ProvidesPackagePass{ghsystem}[2020/02/17]

\ExplSyntaxOn

\cs_set_protected:Npn \ghsystem_filler:n #1
  { \emph { \textless #1 \textgreater } }

\cs_set_protected:Npn \ghsystem_pic:n #1
  {
    \__ghsystem_includegraphics:xn
      {
%         scale = \fp_to_tl:N \l__ghsystem_picture_scale_fp
        width = 1.25cm
        \exp_not:V \l__ghsystem_picture_includegraphics_tl
      }
      { ghsystem_ #1 . \l__ghsystem_picture_type_tl }
  }

\ExplSyntaxOff
```

File 205 **lwarp-gindex.sty**

§ 314 Package gindex

(*Emulates or patches code by* Javier Bezos.)

gindex (*Pkg*) gindex is patched for use by lwarp.

See section 8.6.15.

for HTML output:

```
\LWR@ProvidesPackagePass{gindex}[2019/10/07]
```

Set the index page and range separators. These are set `\AtBeginDocument` to allow the user to change them. They are then protected so that the lwarp core looks for the tokens instead of their expanded contents, since the `*.ind` files will contain `\indexpagessep` and `\indexrangesep` instead of their literal contents. Finally, lwarp is told of the gindex macros.

```
\AtBeginDocument{
    \robustify{\indexpagessep}
    \robustify{\indexrangesep}
    \renewcommand*{\IndexPageSeparator}{\indexpagessep}
    \renewcommand*{\IndexRangeSeparator}{\indexrangesep}
}
```

`\hyperindexref` is added:

```
\def\addindexitem#1#2{%
  \indexflushitem
 \gix@getspecial#1\indexspecial\indexspecial\@@\indexitem{\hyperindexref{#2}}}

\def\addindexsubitem#1#2{%
  \stepcounter{indexsubitems}%
 \gix@getspecial#1\indexspecial\indexspecial\@@\indexsubitem{\hyperindexref{#2}}}

\def\addindexsubsubitem#1#2{%
 \gix@getspecial#1\indexspecial\indexspecial\@@\indexsubsubitem{\hyperindexref{#2}}}
```

Uses a of class indexheading:

```
\renewcommand\indexheading[1]{%
    \begin{BlockClass}{indexheading}
    \MakeUppercase{#1}%
    \end{BlockClass}
}
```

File 206 **lwarp-gloss.sty**

§ 315 Package gloss

(*Emulates or patches code by* Jose Luis Díiaz, Javier Bezos.)

gloss (*Pkg*) gloss is patched for use by lwarp.

To process the HTML glossary:

```
bibtex <projectname>_html.gls
```

for HTML output:

```
\LWR@ProvidesPackagePass{gloss}[2002/07/26]
```

`\BaseJobname` is added to the label in case xr or xr-hyper are used.

```
\xpatchcmd{\gls@gloss@iii}
    {\thepage}
    {\theLWR@previousautopagelabel}
    {}
    {\LWR@patcherror{gloss}{gls@gloss@iii}}

\def\gls@page@i#1#2{%
  \endgroup%
  \global\@namedef{glsp@#1}{\nameref{\BaseJobname-autopage-#2}}}%
```

File 207 **lwarp-glossaries.sty**

§ 316 Package

glossaries

(*Emulates or patches code by* Nicola L.C. Talbot.)

glossaries (*Pkg*)
processing glossaries
GlossaryCmd (*Opt*)
Default: makeglossaries
printglossary (*Opt*) [lwarpmk]
htmlglossary (*Opt*) [lwarpmk]

lwarpmk has the commands **lwarpmk printglossary** and **lwarpmk htmlglossary**, which process the glossaries created by the glossaries package using that package's *makeglossaries* program.

The shell command to execute is set by the lwarp option GlossaryCmd, which defaults to **makeglossaries**. The print or HTML glossary filename is appended to this command.

⚠ *makeglossaries* not found

In some situations it may be required to modify the default command, such as to add the **perl** command in front:

```
\usepackage[
   GlossaryCmd={perl makeglossaries},
] {lwarp}
```

xindy language

To set the language to use for processing glossaries with *xindy*:

```
\usepackage[
   GlossaryCmd={makeglossaries -L english},
] {lwarp}
```

Other options for *makeglossaries* may be set as well.

placement and TOC options

The glossaries may be placed in a numbered or unnumbered section, given a TOC entry, and placed inline or on their own HTML page:

Numbered section, on its own HTML page:

```
\usepackage[xindy,toc,numberedsection=nolabel]{glossaries}
...
\printglossaries
```

Unnumbered section, inline with the current HTML page:

```
\usepackage[xindy,toc]{glossaries}
...
\printglossaries
```

Unnumbered section, on its own HTML page:

```
\usepackage[xindy,toc]{glossaries}
...
\ForceHTMLPage
\printglossaries
```

⚠ glossary style — The default `style=item` option for glossaries conflicts with lwarp, so the style is forced to `index` instead.

⚠ number list — The page number list in the printed form would become `\namerefs` in HTML, which could become a very long string if many items are referenced. For now, the number list is simply turned off.

print/HTML versions — The print and HTML versions of the glossary differ in their internal page numbers. Separate commands for generating print and HTML glossaries are used, even though the page number is currently ignored.

for HTML output:

```
\PassOptionsToPackage{xindy}{glossaries}

\LWR@ProvidesPackagePass{glossaries}[2018/07/23]

\setupglossaries{nonumberlist}
\setglossarystyle{index}
```

Patched to fix TOC pointing to the previous page:

```
\renewcommand*{\@p@glossarysection}[2]{%
  \glsclearpage
  \LWR@phantomsection
  \ifdefempty\@@glossarysecstar
  {%
    \csname\@@glossarysec\endcsname{#2}%
  }%
  {%
```

In the original, the TOC entry was made before the section, thus linking to the phantomsection in the printed version, but for HTML, this caused the link to point to the page before the glossaries, which could be a different HTML file. Here, the TOC entry is made after the section is created:

```
      \csname\@@glossarysec\endcsname*{#2}%
    \@gls@toc{#1}{\@@glossarysec}% Moved after the previous line.
  }%
  \@@glossaryseclabel
}
```

lwarp's sectioning commands cannot handle robust macros when splitting HTML into named filenames. glossaries uses `\translate` in sectioning names, and `\translate` is robust and cannot be expanded. The following pre-expands the translations at this moment, making use of `\translatelet`.

```
\newcommand*{\LWR@comp@glossaryname}{\translate{Glossary}}

\ifdefstrequal{\glossaryname}{\LWR@comp@glossaryname}{
    \translatelet\LWR@translatetemp{Glossary}
    \edef\glossaryname{\LWR@translatetemp}
}{}

\newcommand*{\LWR@comp@acronymname}{\translate{Acronym}}

\ifdefstrequal{\acronymname}{\LWR@comp@acronymname}{
    \translatelet\LWR@translatetemp{Acronym}
    \edef\acronymname{\LWR@translatetemp}
}{}
```

```
33
34 \newcommand*{\LWR@comp@glssymbolsgroupname}{\translate{Symbols (glossaries)}}
35
36 \ifdefstrequal{\glssymbolsgroupname}{\LWR@comp@glssymbolsgroupname}{
37     \translatelet\LWR@translatetemp{Symbols (glossaries)}
38     \edef\glssymbolsgroupname{\LWR@translatetemp}
39 }{}
40
41 \newcommand*{\LWR@comp@glsnumbersgroupname}{\translate{Numbers (glossaries)}}
42
43 \ifdefstrequal{\glsnumbersgroupname}{\LWR@comp@glsnumbersgroupname}{
44     \translatelet\LWR@translatetemp{Numbers (glossaries)}
45     \edef\glsnumbersgroupname{\LWR@translatetemp}
46 }{}
```

File 208 **lwarp-gmeometric.sty**

§ 317 Package gmeometric

gmeometric (*Pkg*) gmeometric is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{gmeometric}[2008/11/22]
2 \RequirePackageWithOptions{geometry}
```

File 209 **lwarp-graphics.sty**

§ 318 Package graphics

(*Emulates or patches code by* D. P. Carlisle.)

graphics (*Pkg*) graphics is emulated.

for HTML output:

```
1 \LWR@ProvidesPackagePass{graphics}[2020/08/30]
```

§ 318.1 Graphics extensions

\DeclareGraphicsExtensions {⟨*list*⟩}

\AtBeginDocument allow SVG files instead of PDF:

```
2 \AtBeginDocument{
3 \DeclareGraphicsExtensions{.svg,.SVG,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}
4 \DeclareGraphicsRule{.svg}{svg}{.svg}{}
5 \DeclareGraphicsRule{.SVG}{svg}{.SVG}{}
6 }
```

Inside a lateximage, allow PDF instead of SVG:

```
7 \ifpdf
8 \appto\LWR@restoreorigformatting{%
9 \DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
10 }
11 \else% \ifpdf
12         \ifXeTeX
```

```
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
}
        \else
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{.eps,.EPS,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
}
        \fi
\fi
```

§ 318.2 Length conversions and graphics options

⚠ whitespace

A scaled image in LaTeX by default takes only as much space on the page as it requires, but HTML browsers use as much space as the original unscaled image would have taken, with the scaled image over- or under-flowing the area.

Used to store the user's selected dimensions and HTML class.

The class defaults to "inlineimage" unless changed by a `class=xyx` option.

```
\newlength{\LWR@igwidth}
\newlength{\LWR@igheight}
\newcommand*{\LWR@igwidthstyle}{}
\newcommand*{\LWR@igheightstyle}{}
\newcommand*{\LWR@igorigin}{}
\newcommand*{\LWR@igangle}{}
\newcommand*{\LWR@igxscale}{1}
\newcommand*{\LWR@igyscale}{1}

\newbool{LWR@igkeepaspectratio}
\boolfalse{LWR@igkeepaspectratio}

\newcommand*{\LWR@igclass}{inlineimage}


\newcommand*{\LWR@igalt}{\ImageAltText}
```

Set the actions of each of the key/value combinations for `\includegraphics`. Many are ignored.

If an optional width was given, set an HTML style:

```
\define@key{igraph}{width}{%
\setlength{\LWR@igwidth}{#1}%
\ifthenelse{\lengthtest{\LWR@igwidth > 0pt}}%
{%
```

Default to use the converted fixed length given:

```
    \renewcommand*{\LWR@igwidthstyle}{width:\LWR@printlength{\LWR@igwidth}}%
```

If ex or em dimensions were given, use those instead:

```
    \IfEndWith{#1}{ex}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes ex
    {}% not ex
    \IfEndWith{#1}{em}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes em
    {}% not em
```

```
    \IfEndWith{#1}{\%}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes percent
    {}% not percent
    \IfEndWith{#1}{px}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes px
    {}% not px
}{}% end of length > 0pt
}
```

If an optional height was given, set an HTML style:

```
\define@key{igraph}{height}{%
\setlength{\LWR@igheight}{#1}%
\ifthenelse{\lengthtest{\LWR@igheight > 0pt}}%
{%
```

Default to use the converted fixed length given:

```
    \renewcommand*{\LWR@igheightstyle}{%
    height:\LWR@printlength{\LWR@igheight} % extra space
    }%
```

If ex or em dimensions were given, use those instead:

```
    \IfEndWith{#1}{ex}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes ex
    {}% not ex
    \IfEndWith{#1}{em}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes em
    {}% not em
    \IfEndWith{#1}{\%}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes percent
    {}% not percent
    \IfEndWith{#1}{px}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes px
    {}% not px
}{}% end of length > 0pt
}
```

Handle `keepaspectratio` key:

```
\define@key{igraph}{keepaspectratio}[false]{%
    \booltrue{LWR@igkeepaspectratio}%
}
```

Handle `origin` key:

```
\define@key{igraph}{origin}[c]{%
    \renewcommand*{\LWR@igorigin}{#1}%
}
```

Handle `angle` key:

```
\define@key{igraph}{angle}{\renewcommand*{\LWR@igangle}{#1}}
```

Handle `class` key:

```
\define@key{igraph}{class}{\renewcommand*{\LWR@igclass}{#1}}
```

Handle `alt` key:

```
\define@key{igraph}{alt}{\renewcommand*{\LWR@igalt}{#1}}
```

It appears that graphicx does not have separate keys for `xscale` and `yscale`. `scale` adjusts both at the same time.

```
\define@key{igraph}{scale}{%
    \ifthenelse{\equal{#1}{1}}{}{%  must expand #1
        \PackageNote{lwarp}{%
        It is recommended to use ``[width=xx\protect\linewidth]''\MessageBreak
            instead of ``[scale=yy]'',%
        }%
    }%
    \renewcommand*{\LWR@igxscale}{#1}%
    \renewcommand*{\LWR@igyscale}{#1}%
}
```

Numerous ignored keys:

```
\define@key{igraph}{bb}{}
\define@key{igraph}{bbllx}{}
\define@key{igraph}{bblly}{}
\define@key{igraph}{bburx}{}
\define@key{igraph}{bbury}{}
\define@key{igraph}{natwidth}{}
\define@key{igraph}{natheight}{}
\define@key{igraph}{hiresbb}[true]{}
\define@key{igraph}{viewport}{}
\define@key{igraph}{trim}{}
\define@key{igraph}{totalheight}{}
\define@key{igraph}{clip}[true]{}
\define@key{igraph}{draft}[true]{}
\define@key{igraph}{type}{}
\define@key{igraph}{ext}{}
\define@key{igraph}{read}{}
\define@key{igraph}{command}{}
```

New in v1.1a:

```
\define@key{igraph}{quite}{}
\define@key{igraph}{page}{}
\define@key{igraph}{pagebox}{}
\define@key{igraph}{interpolate}[true]{}
```

New in v1.1b:

```
\define@key{igraph}{decodearray}{}
```

§ 318.3 Printing HTML styles

\LWR@rotstyle {⟨*prefix*⟩} {⟨*degrees*⟩}

Prints the rotate style with the given prefix.

prefix is -ms- or -webkit- or nothing, and is used to generate three versions of the transform:rotate style.

```
\newcommand*{\LWR@rotstyle}[2]{%
    \edef\LWR@tempone{#2}%
    \setcounter{LWR@tempcountone}{-1*\real{\LWR@tempone}} % space
    #1transform:rotate(\arabic{LWR@tempcountone}deg); % space
}
```

\LWR@scalestyle {⟨*prefix*⟩} {⟨*xscale*⟩} {⟨*yscale*⟩}

Prints the scale style with the given prefix.

prefix is -ms- or -webkit- or nothing, and is used to generate three versions of the transform:scale style.

```
\newcommand*{\LWR@scalestyle}[3]{%
    #1transform:scale(#2,#3);
}
```

§ 318.4 \includegraphics

\LWR@opacity For HTML, used only for \includegraphics.

\LWR@opacity may be set by the transparent package.

```
\def\LWR@opacity{1}
```

\LWR@imagesizebox Used to determine the actual image size if needed.

```
\newsavebox{\LWR@imagesizebox}
```

\LWR@HTML@Gin@setfile {⟨*w*⟩} {⟨*h*⟩} {⟨*filename*⟩} Sets the parsed filename for HTML output.

```
\newcommand*{\LWR@HTML@Gin@setfile}[3]{%
    \xdef\LWR@parsedfilename{#3}%
}
```

class (*Key*) [Gin] CSS class for the image.

Define the new class key for the print-mode version of \includegraphics, which is enabled inside a lateximage.

```
\AtBeginDocument{
\define@key{Gin}{class}{}
}
```

\LWR@replaceEPSSVG

Usually, references to EPS files become SVG files, but if the epstopdf package is being used, it automatically converts EPS to PDF, and the following must NOT be done.

```
\AtBeginDocument{
\IfPackageLoadedTF{epstopdf}
{
    \newcommand*{\LWR@replaceEPSSVG}{}
}{%
```

```
    \newcommand*{\LWR@replaceEPSSVG}{%
        \StrSubstitute{\LWR@tempone}{.eps}{.svg}[\LWR@tempone]%
        \StrSubstitute{\LWR@tempone}{.EPS}{.SVG}[\LWR@tempone]%
    }
}%
}
```

* [⟨*2: options*⟩] [⟨*3: options*⟩] {⟨*4: filename*⟩}

\LWR@ig@useactualimagesize

If formatting for a word processor, find and set the actual image size, without rotation, using PDF instead of SVG to find the original bounding box:

```
\newcommand*{\LWR@ig@useactualimagesize}[4]{%
    \begingroup%
    \LWR@restoreorigformatting%
    \ifpdf%
    \appto\LWR@restoreorigformatting{%
        \DeclareGraphicsExtensions{%
            .pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
        }%
    }%
    \else% \ifpdf
            \ifXeTeX%
    \appto\LWR@restoreorigformatting{%
        \DeclareGraphicsExtensions{%
            .pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
        }%
    }%
            \else%
    \appto\LWR@restoreorigformatting{%
        \DeclareGraphicsExtensions{%
            .eps,.EPS,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
        }%
    }%
            \fi%
    \fi% \ifpdf
```

For a word processor, do not use rotation:

```
    \ifbool{FormatWP}{\define@key{Gin}{angle}{}}{}%

    \IfBooleanTF{#1}%
    {% starred
        \IfValueTF{#3}%
        {%
            \global\sbox{\LWR@imagesizebox}{%
                \LWR@originincludegraphics*[#2][#3]{#4}%
            }%
        }%
        {%
            \IfValueTF{#2}%
            {%
                \global\sbox{\LWR@imagesizebox}{%
                    \LWR@originincludegraphics*[#2]{#4}%
                }%
            }{%
                \global\sbox{\LWR@imagesizebox}{%
                    \LWR@originincludegraphics*{#4}%
                }%
            }%
        }%
    }% starred
```

```
    {% not starred
        \IfValueTF{#3}%
        {%
            \global\sbox{\LWR@imagesizebox}{%
                \LWR@origincludegraphics[#2][#3]{#4}%
            }%
        }%
        {%
            \IfValueTF{#2}%
            {%
                \global\sbox{\LWR@imagesizebox}{%
                    \LWR@origincludegraphics[#2]{#4}%
                }%
            }{%
                \global\sbox{\LWR@imagesizebox}{%
                    \LWR@origincludegraphics{#4}%
                }%
            }%
        }%
    }% not starred
    \endgroup%
    \settowidth{\LWR@igwidth}{\usebox{\LWR@imagesizebox}}%
    \global\renewcommand*{\LWR@igwidthstyle}{%
        width:\LWR@printlength{\LWR@igwidth}%
    }%
    \settoheight{\LWR@igheight}{\usebox{\LWR@imagesizebox}}%
    \global\renewcommand*{\LWR@igheightstyle}{%
        height:\LWR@printlength{\LWR@igheight}%
    }%
}
```

\LWR@ig@htmltag

For the HTML reference, add the graphicspath, filename, extension, alt tag, style, and class.

```
\newcommand*{\LWR@ig@htmltag}{%
    img\LWR@indentHTML%
    src=\textquotedbl%
    \detokenize\expandafter{\LWR@parsedfilename}%
    \textquotedbl\LWR@indentHTML%
```

Only include a style tag if a width, height, angle, or scale was given:

```
    \ifthenelse{
        \NOT\equal{\LWR@igwidthstyle}{} \OR
        \NOT\equal{\LWR@igheightstyle}{} \OR
        \NOT\equal{\LWR@igorigin}{} \OR
        \NOT\equal{\LWR@igangle}{} \OR
        \NOT\equal{\LWR@igxscale}{1} \OR
        \NOT\equal{\LWR@igyscale}{1}
    }%
    {%
        style=\textquotedbl\LWR@indentHTML
        \ifthenelse{\NOT\equal{\LWR@igwidthstyle}{}}%
            {\LWR@igwidthstyle;\LWR@indentHTML}{}%
        \ifthenelse{\NOT\equal{\LWR@igheightstyle}{}}%
            {\LWR@igheightstyle;\LWR@indentHTML}{}%
        \ifthenelse{\NOT\equal{\LWR@igorigin}{}}%
            {%
                transform-origin: \LWR@originnames{\LWR@igorigin};%
```

```
                \LWR@indentHTML%
            }{}%
        \ifthenelse{\NOT\equal{\LWR@igangle}{}}%
        {%
            \LWR@rotstyle{-ms-}{\LWR@igangle}\LWR@indentHTML
            \LWR@rotstyle{-webkit-}{\LWR@igangle}\LWR@indentHTML
            \LWR@rotstyle{}{\LWR@igangle }\LWR@indentHTML
        }{}%
        \ifthenelse{%
            \NOT\equal{\LWR@igxscale}{1}\OR%
            \NOT\equal{\LWR@igyscale}{1}%
        }%
        {%
            \LWR@scalestyle{-ms-}{\LWR@igxscale}{\LWR@igyscale}%
            \LWR@indentHTML
            \LWR@scalestyle{-webkit-}{\LWR@igxscale}{\LWR@igyscale}%
            \LWR@indentHTML
            \LWR@scalestyle{}{\LWR@igxscale}{\LWR@igyscale}%
            \LWR@indentHTML
        }{}%
        %
        \ifthenelse{\NOT\equal{\LWR@opacity}{1}}%
            {opacity:\LWR@opacity;\LWR@indentHTML}{}%
        %
        \textquotedbl\LWR@indentHTML%
    }{}%
```

Set the class and alt tag:

```
    class=\textquotedbl\LWR@igclass\textquotedbl\LWR@indentHTML%
  alt=\textquotedbl\AltTextOpen\LWR@igalt\AltTextClose\textquotedbl\ \LWR@orignewline%
}% end of image tags
```

\LWR@includegraphicsb * [⟨*2: options*⟩] [⟨*3: options*⟩] {⟨*4: filename*⟩}

graphics syntax is \includegraphics * [⟨*llx,lly*⟩] [⟨*urx,ury*⟩] {⟨*filename*⟩}

graphicx syntax is \includegraphics [⟨*key values*⟩] {⟨*filename*⟩}

If #3 is empty, only one optional argument was given, thus graphicx syntax.

If using \epsfig or \psfig from the epsfig package, #4 will be \LWR@epsfig@filename, which will have been set by the file or figure keys. Therefore, #4 must not be used until after the keys have been processed.

```
\NewDocumentCommand{\LWR@includegraphicsb}{s o o m}
{%
```

Start the image tag on a new line, allow PDF output word wrap:

```
    \LWR@origtilde \LWR@orignewline%
```

Temporarily compute \linewidth, \textwidth, \textheight arguments with a 6x9 inch size until the next \endgroup.

```
    \begin{LWR@setvirtualpage}%
```

For correct em sizing during the width and height conversions:

```
    \large%
```

Temporarily prevent underfull \hbox warnings.

```
    \hbadness=10000\relax%
```

Reset some defaults, possibly will be changed below if options were given:

```
    \setlength{\LWR@igwidth}{0pt}%
    \setlength{\LWR@igheight}{0pt}%
    \renewcommand*{\LWR@igwidthstyle}{}%
    \renewcommand*{\LWR@igheightstyle}{}%
    \renewcommand*{\LWR@igorigin}{}%
    \renewcommand*{\LWR@igangle}{}%
    \renewcommand*{\LWR@igxscale}{1}%
    \renewcommand*{\LWR@igyscale}{1}%
    \renewcommand*{\LWR@igclass}{inlineimage}%
    \boolfalse{LWR@igkeepaspectratio}%

    \ifdefvoid{\LWR@ThisAltText}{%
        \edef\LWR@igalt{\ImageAltText}%
    }{%
        \edef\LWR@igalt{\LWR@ThisAltText}%
    }%
```

If `#3` is empty, only one optional argument was given, thus graphicx syntax:

```
    \IfValueF{#3}{%
        \IfValueTF{#2}%
            {\setkeys{igraph}{#2}}%
            {\setkeys{igraph}{}}%
    }%
```

Fully expand and detokenize the filename, changing the file extension to `.svg` if necessary.

Note that uppercase file extensions are detected and reported as lowercase, so lwarp can only report to the browser lowercase extensions, so all images must have lowercase file extensions.

```
    \begingroup%
    \LetLtxMacro\Gin@setfile\LWR@HTML@Gin@setfile%
    \edef\LWR@tempone{#4}%
```

PDF extensions are removed to allow a search for another graphics format such as SVG or PNG.

```
    \StrSubstitute{\LWR@tempone}{.pdf}{}[\LWR@tempone]%
    \StrSubstitute{\LWR@tempone}{.PDF}{}[\LWR@tempone]%

    \LWR@replaceEPSSVG%
    \xdef\LWR@parsedfilename{\LWR@tempone}%
    \Ginclude@graphics{\detokenize\expandafter{\LWR@parsedfilename}}%
    \endgroup%
    \filename@parse{\LWR@parsedfilename}%
```

Remove doubled `//` in the directory path, from the 2020/10/01 LaTeX kernel change.

```
    \StrSubstitute{\LWR@parsedfilename}{//}{/}[\LWR@parsedfilename]%

    \LWR@traceinfo{LWR@parsedfilename is \LWR@parsedfilename}%
```

If formatting for a word processor, or if using `keepaspectratio`, find and set the actual image size, without rotation, using PDF instead of SVG to find the original bounding box:

```
    \ifboolexpr{
        bool {FormatWP} or
        bool {LWR@igkeepaspectratio}
    }{\LWR@ig@useactualimagesize{#1}{#2}{#3}{#4}}{}%
```

Create the HTML reference with the graphicspath, filename, extension, alt tag, style, and class:

```
    \LWR@traceinfo{LWR@includegraphicsb: about to create href}%
    \LWR@href{\LWR@parsedfilename}%
    {% start of href
        \LWR@traceinfo{LWR@includegraphicsb: about to LWR@htmltag}%
        \LWR@htmltag{\LWR@ig@htmltag}%
    }% end of href
```

Return to original page size and font size:

```
    \end{LWR@setvirtualpage}%
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
    \LWR@traceinfo{LWR@includegraphicsb done}%
}
```

`\includegraphics` `[`⟨*key=val*⟩`]` `{`⟨*filename*⟩`}`

Handles width and height, converted to fixed width and heights.

The user should always use no file suffix in the document source.

```
\AtBeginDocument{

\LWR@traceinfo{Patching includegraphics.}

\LetLtxMacro\LWR@origincludegraphics\includegraphics


\renewrobustcmd*{\includegraphics}
{%
```

This graphic should trigger an HTML paragraph even if alone, so ensure that are doing paragraph handling:

```
\LWR@traceinfo{includegraphics}%
\LWR@ensuredoingapar%
\LWR@includegraphicsb%
}% includegraphics
}% AtBeginDocument
```

§ 318.5 Boxes

`\LWR@rotboxorigin` Holds the origin key letters.

```
\newcommand*{\LWR@rotboxorigin}{}
```

`\LWR@originname` `{`⟨*letter*⟩`}`

Given one LaTeX origin key value, translate into an HTML origin word:

```
\newcommand*{\LWR@originname}[1]{%
    \ifthenelse{\equal{#1}{t}}{top}{}%
    \ifthenelse{\equal{#1}{b}}{bottom}{}%
    \ifthenelse{\equal{#1}{c}}{center}{}%
    \ifthenelse{\equal{#1}{l}}{left}{}%
    \ifthenelse{\equal{#1}{r}}{right}{}%
}
```

\LWR@originnames {⟨*letters*⟩}

Given one- or two-letter LaTeX origin key values, translate into HTML origin words:

```
\newcommand*{\LWR@originnames}[1]{%
\StrChar{#1}{1}[\LWR@strresult]%
\LWR@originname{\LWR@strresult}
\StrChar{#1}{2}[\LWR@strresult]%
\LWR@originname{\LWR@strresult}
}
```

Handle the `origin` key for `\rotatebox`:

```
\define@key{krotbox}{origin}{%
\renewcommand*{\LWR@rotboxorigin}{#1}%
}
```

These keys are ignored:

```
\define@key{krotbox}{x}{}
\define@key{krotbox}{y}{}
\define@key{krotbox}{units}{}
```

\rotatebox [⟨*keyval list*⟩] {⟨*angle*⟩} {⟨*text*⟩}

```
\AtBeginDocument{
```

The HTML version:

```
\NewDocumentCommand{\LWR@HTML@rotatebox}{O{} m +m}{%
```

Reset the origin to “none-given”:

```
\renewcommand*{\LWR@rotboxorigin}{}
```

Process the optional keys, which may set `\LWR@rotateboxorigin`:

```
\setkeys{krotbox}{#1}%
```

Select `inline-block` so that HTML will transform this span:

```
\LWR@htmltagc{%
    span\LWR@indentHTML
    style=\textquotedbl\LWR@indentHTML
    display: inline-block;\LWR@indentHTML
```

If an origin was given, translate and print the origin information:

```
    \ifthenelse{\NOT\equal{\LWR@rotboxorigin}{}}%
     {transform-origin: \LWR@originnames{\LWR@rotboxorigin};\LWR@indentHTML}%
        {}%
```

Print the rotation information:

```
    \LWR@rotstyle{-ms-}{#2}\LWR@indentHTML
    \LWR@rotstyle{-webkit-}{#2}\LWR@indentHTML
    \LWR@rotstyle{}{#2}\textquotedbl\LWR@orignewline%
}\LWR@orignewline%
```

Print the text to be rotated:

```
\begin{LWR@nestspan}%
#3%
```

Close the span:

```
\LWR@htmltagc{/span}%
\end{LWR@nestspan}%
}
```

The high-level interface:

```
\LWR@formatted{rotatebox}

}% AtBeginDocument
```

`\scalebox` {⟨*h-scale*⟩} [⟨*v-scale*⟩] {⟨*text*⟩}

```
\AtBeginDocument{
```

The HTML version:

```
\NewDocumentCommand{\LWR@HTML@scalebox}{m o m}{%
```

Select `inline-block` so that HTML will transform this span:

```
\LWR@htmltagc{%
    span\LWR@indentHTML
    style=\textquotedbl\LWR@indentHTML
    display: inline-block;\LWR@indentHTML
```

Print the scaling information:

```
    \LWR@scalestyle{-ms-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
    \LWR@scalestyle{-webkit-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
    \LWR@scalestyle{}{#1}{\IfNoValueTF{#2}{#1}{#2}}
    \textquotedbl\LWR@orignewline
}\LWR@orignewline%
```

Print the text to be scaled:

```
\begin{LWR@nestspan}%
#3%
```

Close the span:

```
\LWR@htmltagc{/span}%
\end{LWR@nestspan}%
}
```

The high-level interface:

```
\LWR@formatted{scalebox}

}% AtBeginDocument
```

`\reflectbox` {⟨*text*⟩}

```
\AtBeginDocument{

\newcommand{\LWR@HTML@reflectbox}[1]{%
    \scalebox{-1}[1]{#1}%
}% \reflectbox

\LWR@formatted{reflectbox}

}% AtBeginDocument
```

`\resizebox` {⟨*h-length*⟩} {⟨*v-length*⟩} {⟨*text*⟩}

Simply prints its text argument.

```
\AtBeginDocument{

\NewDocumentCommand{\LWR@HTML@resizebox}{s m m m}{%
    #4%
}

\LWR@formatted{resizebox}

}% AtBeginDocument
```

File 210 **lwarp-graphicx.sty**

§ 319 Package graphicx

`graphicx` (*Pkg*) graphicx is emulated.

graphicx loads graphics, which also loads lwarp-graphics, which remembers the original graphics definitions for use inside a `latexmage`, and then patches them `\AtBeginDocument` for HTML output.

lwarp-graphics handles the syntax of either graphics or graphicx.

for HTML output:

```
\LWR@ProvidesPackagePass{graphicx}[2020/09/09]
```

File 211 **lwarp-grffile.sty**

§ 320 Package grffile

`grffile` (*Pkg*) grffile is supported as-is. File types known to the browser are displayed, and unknown file types are given a link. Each PDF image for print mode should be

⚠ matching PDF and SVG accompanied by an SVG, PNG, or JPG version for HTML.

lwarp-grffile now exists as a placeholder since grffile used to be emulated by lwarp, and thus older versions of lwarp-grffile may exist and should be overwritten by this newer version.

for HTML output:

```
\LWR@ProvidesPackagePass{grffile}[2017/06/30]
```

File 212 **lwarp-grid.sty**

§ 321 Package grid

grid (*Pkg*) grid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{grid}[2009/06/16]

\newenvironment*{gridenv}{}{}
```

File 213 **lwarp-grid-system.sty**

§ 322 Package grid-system

(*Emulates or patches code by* Marcus Bitzl.)

grid-system (*Pkg*) grid-system is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{grid-system}[2014/02/16]
```

(\ifdef is in case the older syntax is removed.)

```
\AtBeginEnvironment{Row}{\setlength{\linewidth}{6in}}

\ifdef{\endrow}{
    \AtBeginEnvironment{row}{\setlength{\linewidth}{6in}}
}{}

\renewcommand{\gridsystem@finishcell}{\hspace{\gridsystem@cellsep}}
```

File 214 **lwarp-gridset.sty**

§ 323 Package gridset

gridset (*Pkg*) gridset is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{gridset}[2020-02-12]

\newcommand*{\gridbase}{}
\newcommand*{\gridinterval}{}
\newcommand*{\SavePos}[1]{}
\ifLuaTeX
\else
\let\savepos\SavePos
\fi
\newcommand*{\vskipnextgrid}{}
\newcommand*{\thegridinfo}[1]{(thegridinfo)}
\newcommand*{\theposinfo}[1]{(theposinfo)}
\newcommand*{\theypos}[1]{(theypos)}
```

File 215 **lwarp-hang.sty**

§ 324 Package

hang

(*Emulates or patches code by* Andreas Nolda.)

hang (*Pkg*) hang is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{hang}[2017/02/18]

\newlength{\hangingindent}
\setlength{\hangingindent}{1em}
\newlength{\hangingleftmargin}
\setlength{\hangingleftmargin}{0em}

\newcommand*{\LWR@findhangingleftmargin}{%
\setlength{\LWR@templengthone}{\hangingleftmargin}%
\addtolength{\LWR@templengthone}{\hangingindent}%
}

\newenvironment{hangingpar}
{
    \LWR@findhangingleftmargin%
    \BlockClass[%
       \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
        \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
    ]%
    {hangingpar}%
}
{\endBlockClass}

\newenvironment{hanginglist}
{%
    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
    \renewcommand*{\LWR@printopenlist}{%
        \LWR@findhangingleftmargin%
        ul style=\textquotedbl%
            \LWR@print@mbox{list-style-type:none;} % extra space
            \LWR@print@mbox{%
                margin-left:\LWR@printlength{\LWR@templengthone}%
            } ; % extra space
            \LWR@print@mbox{%
                text-indent:-\LWR@printlength{\hangingindent}%
            }%
        \textquotedbl%
    }%
    \LetLtxMacro\item\LWR@itemizeitem%
    \list{}{}%
}
{\endlist}

\newenvironment{compacthang}
{\hanginglist}
{\endhanginglist}

\newlength{\labeledleftmargin}
```

```
\setlength{\labeledleftmargin}{0em}

\newenvironment{labeledpar}[2]
{%
    \BlockClass[%
        \LWR@findhangingleftmargin%
       \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
        \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
    ]{labeledpar}#2%
}
{\endBlockClass}

\newenvironment{labeledlist}[1]
{\hanginglist}
{\endhanginglist}

\newenvironment{compactlabel}[1]
{\hanginglist}
{\endhanginglist}
```

File 216 **lwarp-hanging.sty**

§ 325 Package

hanging

hanging (*Pkg*) hanging is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{hanging}[2009/09/02]

\IfClassLoadedTF{memoir}{
\let\hangpara\relax
\let\hangparas\relax
\let\endhangparas\relax
\let\hangpunct\relax
\let\endhangpunct\relax
}{}
```

\hangpara {⟨*indent*⟩} {⟨*afternum*⟩}

Use hangparas instead.

```
\newcommand*{\hangpara}[2]{}
```

Env hangparas {⟨*indent*⟩} {⟨*afternum*⟩}

```
\newenvironment*{hangparas}[2]
{%
    \BlockClass[%
        \LWR@print@mbox{margin-left:\LWR@printlength{#1}} ; %
        \LWR@print@mbox{text-indent:-\LWR@printlength{#1}}%
    ]%
    {hangingpar}%
}
{\endBlockClass}
```

Env hangpunct

```
\newenvironment*{hangpunct}
{\BlockClass{hangpunct}}
{\endBlockClass}

\newcommand{\nhpt}{.}
\newcommand{\nhlq}{‘}
\newcommand{\nhrq}{’}
```

File 217 **lwarp-hepunits.sty**

§ 326 Package

hepunits

(*Emulates or patches code by* Andy Buckley.)

hepunits (*Pkg*) hepunits is used as-is, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{hepunits}[2020/04/10]

\begin{warpMathJax}
\LWR@infoprocessingmathjax{hepunits}

\ifx\@HEPopt@sicmds\@yes
\CustomizeMathJax{\newcommand{\micron}{\micro\metre}}
\CustomizeMathJax{\newcommand{\mrad}{\milli\radian}}
\fi

\CustomizeMathJax{\newcommand{\gauss}{\mathrm{G}}}

\CustomizeMathJax{\newcommand{\invcmsq}{\centi\metre\tothe{-2}}}
\CustomizeMathJax{\newcommand{\invcmsqpersecond}{\invcmsq\second\tothe{-1}}}
\CustomizeMathJax{\newcommand{\invcmsqpersec}{\invcmsqpersecond}}

%% (Inverse) cross-sections
\CustomizeMathJax{\newcommand{\invbarn}{\barn\tothe{-1}}}

\ifx\@HEPopt@noprefixcmds\@empty
\CustomizeMathJax{\newcommand{\millibarn}{\milli\barn}}
\CustomizeMathJax{\newcommand{\microbarn}{\micro\barn}}
\CustomizeMathJax{\newcommand{\nanobarn}{\nano\barn}}
\CustomizeMathJax{\newcommand{\picobarn}{\pico\barn}}
\CustomizeMathJax{\newcommand{\femtobarn}{\femto\barn}}
\CustomizeMathJax{\newcommand{\attobarn}{\atto\barn}}
\CustomizeMathJax{\newcommand{\zeptobarn}{\zepto\barn}}
\CustomizeMathJax{\newcommand{\yoctobarn}{\yocto\barn}}
\CustomizeMathJax{\newcommand{\invnanobarn}{\nano\invbarn}}
\CustomizeMathJax{\newcommand{\invpicobarn}{\pico\invbarn}}
\CustomizeMathJax{\newcommand{\invfemtobarn}{\femto\invbarn}}
\CustomizeMathJax{\newcommand{\invattobarn}{\atto\invbarn}}
\CustomizeMathJax{\newcommand{\invzeptobarn}{\zepto\invbarn}}
\CustomizeMathJax{\newcommand{\invyoctobarn}{\yocto\invbarn}}
\CustomizeMathJax{\newcommand{\invnb}{\invnanobarn}}
\CustomizeMathJax{\newcommand{\invpb}{\invpicobarn}}
\CustomizeMathJax{\newcommand{\invfb}{\invfemtobarn}}
\CustomizeMathJax{\newcommand{\invab}{\invattobarn}}
\CustomizeMathJax{\newcommand{\invzb}{\invzeptobarn}}
\CustomizeMathJax{\newcommand{\invyb}{\invyoctobarn}}
\fi
```

```
41
42 \CustomizeMathJax{\newcommand{\electronvoltc}{\electronvolt\per\mathit{c}}}
43 \CustomizeMathJax{\newcommand{\electronvoltcsq}{\electronvolt\per\mathit{c}\squared}}
44 \CustomizeMathJax{\let\eVc\electronvoltc}
45 \CustomizeMathJax{\let\eVcsq\electronvoltcsq}
46
47 \ifx\@HEPopt@noprefixcmds\@empty
48 \CustomizeMathJax{\newcommand{\meV}{\milli\eV}}
49 \CustomizeMathJax{\newcommand{\keV}{\kilo\eV}}
50 \CustomizeMathJax{\newcommand{\MeV}{\mega\eV}}
51 \CustomizeMathJax{\newcommand{\GeV}{\giga\eV}}
52 \CustomizeMathJax{\newcommand{\TeV}{\tera\eV}}
53 \CustomizeMathJax{\newcommand{\meVc}{\milli\eVc}}
54 \CustomizeMathJax{\newcommand{\keVc}{\kilo\eVc}}
55 \CustomizeMathJax{\newcommand{\MeVc}{\mega\eVc}}
56 \CustomizeMathJax{\newcommand{\GeVc}{\giga\eVc}}
57 \CustomizeMathJax{\newcommand{\TeVc}{\tera\eVc}}
58 \CustomizeMathJax{\newcommand{\meVcsq}{\milli\eVcsq}}
59 \CustomizeMathJax{\newcommand{\keVcsq}{\kilo\eVcsq}}
60 \CustomizeMathJax{\newcommand{\MeVcsq}{\mega\eVcsq}}
61 \CustomizeMathJax{\newcommand{\GeVcsq}{\giga\eVcsq}}
62 \CustomizeMathJax{\newcommand{\TeVcsq}{\tera\eVcsq}}
63 \fi
64 \end{warpMathJax}
```

File 218 **lwarp-hhline.sty**

§ 327 Package

hhline

(*Emulates or patches code by* David Carlisle.)

hhline (*Pkg*) hhline is patched for use by lwarp.

Only a rudimentary emulation is provided so far. If the argument contains any = characters, the result is a double \hline. If none, the result is a single \hline.

for HTML output:

```
1 \LWR@ProvidesPackagePass{hhline}[2014/10/28]

2 \newrobustcmd*{\LWR@HTML@hhline}[1]{%
3     \edef\LWR@tempone{\detokenize\expandafter{#1}}%
4     \IfSubStr[1]{\LWR@tempone}{=}{\hline\hline}{\hline}%
5 }
6 % ^^A or:
7 % ^^A \newrobustcmd*{\LWR@HTML@hhline}[1]{\LWR@getmynexttoken}
8
9 \AtBeginDocument{\LWR@expandableformatted{hhline}}
```

For MathJax. A simple \hline is used.

```
10 \begin{warpMathJax}
11 \CustomizeMathJax{\newcommand{\hhline}[1]{\hline}}
12 \end{warpMathJax}
```

File 219 **lwarp-hhtensor.sty**

§ 328 Package hhtensor

(*Emulates or patches code by* Harald Harders.)

hhtensor (*Pkg*) hhtensor is used as-is, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{hhtensor}[2011/12/29]

\begin{warpMathJax}
\iftensor@bold
  \CustomizeMathJax{\newcommand{\vec}[1]{\boldsymbol{#1}}}
  \CustomizeMathJax{\newcommand{\matr}[1]{\boldsymbol{#1}}}
  \CustomizeMathJax{\newcommand{\tens}[2]{\boldsymbol{#1}}}
\else
  \iftensor@uline
    \CustomizeMathJax{\newcommand{\vec}[1]{\ushort{#1}}}
    \CustomizeMathJax{\newcommand{\matr}[1]{\ushortd{#1}}}
    \CustomizeMathJax{\newcommand{\tens}[2]{
        \underset{
            \raise{.5ex}{\underset{#2}{\sim}}
        }{#1}
    }}
  \else
    \CustomizeMathJax{\newcommand{\matr}[1]{\vec{\vec{#1}}}}
    \CustomizeMathJax{\newcommand{\tens}[2]{
        \underset{
            \raise{.5ex}{\underset{#2}{\sim}}
        }{#1}
    }}
  \fi
\fi
\CustomizeMathJax{\newcommand{\dcdot}{\mathrel{\cdot\mkern 0.0mu \cdot}}}
\CustomizeMathJax{\newcommand{\trans}{{}^{\mathrm{T}}}}
\end{warpMathJax}
```

File 220 **lwarp-hypbmsec.sty**

§ 329 Package hypbmsec

hypbmsec (*Pkg*) hypbmsec is emulated by the lwarp core.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypbmsec}[2016/05/16]
```

File 221 **lwarp-hypcap.sty**

§ 330 Package hypcap

hypcap (*Pkg*) hypcap is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypcap}[2016/05/16]

\newcommand*{\capstart}{}
\newcommand*{\hypcapspace}{}
\newcommand*{\hypcapredef}[1]{}
\newcommand*{\capstartfalse}{}
\newcommand*{\capstarttrue}{}
```

File 222 **lwarp-hypdestopt.sty**

§ 331 Package hypdestopt

hypdestopt (*Pkg*) hypdestopt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypdestopt}[2016/05/21]
```

File 223 **lwarp-hypernat.sty**

§ 332 Package hypernat

hypernat (*Pkg*) hypernat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypernat}[2001/07/09]
```

File 224 **lwarp-hyperref.sty**

§ 333 Package hyperref

(*Emulates or patches code by* Sebastian Rahtz, Heiko Oberdiek, The LaTeX3 Project.)

hyperref (*Pkg*) hyperref is emulated.

for HTML output:

```
% \LWR@ProvidesPackageDrop{hyperref}% not allowed
% \ProvidesPackage{lwarp-#1-#2}% not allowed
\PackageInfo{lwarp}{%
Using the lwarp HTML version of package ‘hyperref’,\MessageBreak
and discarding options except backref, pagebackref.\MessageBreak
(Not using \protect\ProvidesPackage, so that other packages\MessageBreak
do not attempt to patch lwarp’s version of ‘hyperref’.)\MessageBreak}

\SetupKeyvalOptions{family=LWR@hyperref,prefix=LWR@hyperref@}

\newcommand{\hypersetup}[1]{\setkeys{LWR@hyperref}{#1}}

\define@key{LWR@hyperref}{a4paper}[]{}
\define@key{LWR@hyperref}{a5paper}[]{}
\define@key{LWR@hyperref}{b5paper}[]{}
\define@key{LWR@hyperref}{letterpaper}[]{}
\define@key{LWR@hyperref}{legalpaper}[]{}
\define@key{LWR@hyperref}{executivepaper}[]{}
\define@key{LWR@hyperref}{implicit}[]{}
```

```
\define@key{LWR@hyperref}{draft}[]{}
\define@key{LWR@hyperref}{final}[]{}
\define@key{LWR@hyperref}{setpagesize}[]{}
\define@key{LWR@hyperref}{debug}[]{}
\define@key{LWR@hyperref}{linktocpage}[]{}
\define@key{LWR@hyperref}{linktoc}[]{}
\define@key{LWR@hyperref}{extension}[]{}
\define@key{LWR@hyperref}{verbose}[]{}
\define@key{LWR@hyperref}{typexml}[]{}
\define@key{LWR@hyperref}{raiselinks}[]{}
\define@key{LWR@hyperref}{breaklinks}[]{}
\define@key{LWR@hyperref}{localanchorname}[]{}
\define@key{LWR@hyperref}{pageanchor}[]{}
\define@key{LWR@hyperref}{plainpages}[]{}
\define@key{LWR@hyperref}{naturalnames}[]{}
\define@key{LWR@hyperref}{hypertexnames}[]{}
\define@key{LWR@hyperref}{nesting}[]{}
\define@key{LWR@hyperref}{destlabel}[]{}
\define@key{LWR@hyperref}{unicode}[]{}
\define@key{LWR@hyperref}{pdfencoding}[]{}
\define@key{LWR@hyperref}{psdextra}[]{}
\define@key{LWR@hyperref}{pdfversion}[]{}
\define@key{LWR@hyperref}{dvipdfmx-outline-open}[]{}
\define@key{LWR@hyperref}{driverfallback}[]{}
\define@key{LWR@hyperref}{customdriver}[]{}
\define@key{LWR@hyperref}{hyperfigures}[]{}
\define@key{LWR@hyperref}{hyperfootnotes}[]{}
\define@key{LWR@hyperref}{hyperindex}[]{}
\define@key{LWR@hyperref}{encap}[]{}
\define@key{LWR@hyperref}{colorlinks}[]{}
\define@key{LWR@hyperref}{ocgcolorlinks}[]{}
\define@key{LWR@hyperref}{frenchlinks}[]{}
\define@key{LWR@hyperref}{bookmarks}[]{}
\define@key{LWR@hyperref}{bookmarksopen}[]{}
\define@key{LWR@hyperref}{bookmarksdepth}[]{}
\define@key{LWR@hyperref}{bookmarksopenlevel}[]{}
\define@key{LWR@hyperref}{bookmarkstype}[]{}
\define@key{LWR@hyperref}{bookmarksnumbered}[]{}
\define@key{LWR@hyperref}{CJKbookmarks}[]{}
\define@key{LWR@hyperref}{link}[]{}
\define@key{LWR@hyperref}{anchor}[]{}
\define@key{LWR@hyperref}{cite}[]{}
\define@key{LWR@hyperref}{file}[]{}
\define@key{LWR@hyperref}{url}[]{}
\define@key{LWR@hyperref}{menu}[]{}
\define@key{LWR@hyperref}{run}[]{}
\define@key{LWR@hyperref}{linkbordercolor}[]{}
\define@key{LWR@hyperref}{anchorbordercolor}[]{}
\define@key{LWR@hyperref}{citebordercolor}[]{}
\define@key{LWR@hyperref}{filebordercolor}[]{}
\define@key{LWR@hyperref}{urlbordercolor}[]{}
\define@key{LWR@hyperref}{menubordercolor}[]{}
\define@key{LWR@hyperref}{runbordercolor}[]{}
\define@key{LWR@hyperref}{pagecolor}[]{}
\define@key{LWR@hyperref}{baseurl}[]{}
\define@key{LWR@hyperref}{linkfileprefix}[]{}
\define@key{LWR@hyperref}{pdfpagetransition}[]{}
\define@key{LWR@hyperref}{pdfpageduration}[]{}
\define@key{LWR@hyperref}{pdfpagehidden}[]{}
\define@key{LWR@hyperref}{pagebordercolor}[]{}
```

```
\define@key{LWR@hyperref}{allbordercolors}[]{}
\define@key{LWR@hyperref}{pdfhighlight}[]{}
\define@key{LWR@hyperref}{pdfborder}[]{}
\define@key{LWR@hyperref}{pdfborderstyle}[]{}
\define@key{LWR@hyperref}{pdfprintpagerange}[]{}
\define@key{LWR@hyperref}{pdfusetitle}[]{}
\define@key{LWR@hyperref}{pdftitle}[]{}
\define@key{LWR@hyperref}{pdfauthor}[]{}
\define@key{LWR@hyperref}{pdfproducer}[]{}
\define@key{LWR@hyperref}{pdfcreator}[]{}
\define@key{LWR@hyperref}{addtopdfcreator}[]{}
\define@key{LWR@hyperref}{pdfcreationdate}[]{}
\define@key{LWR@hyperref}{pdfmoddate}[]{}
\define@key{LWR@hyperref}{pdfsubject}[]{}
\define@key{LWR@hyperref}{pdfkeywords}[]{}
\define@key{LWR@hyperref}{pdftrapped}[]{}
\define@key{LWR@hyperref}{pdfinfo}[]{}
\define@key{LWR@hyperref}{pdfview}[]{}
\define@key{LWR@hyperref}{pdflinkmargin}[]{}
\define@key{LWR@hyperref}{pdfstartpage}[]{}
\define@key{LWR@hyperref}{pdfstartview}[]{}
\define@key{LWR@hyperref}{pdfremotestartview}[]{}
\define@key{LWR@hyperref}{pdfpagescrop}[]{}
\define@key{LWR@hyperref}{pdftoolbar}[]{}
\define@key{LWR@hyperref}{pdfmenubar}[]{}
\define@key{LWR@hyperref}{pdfwindowui}[]{}
\define@key{LWR@hyperref}{pdffitwindow}[]{}
\define@key{LWR@hyperref}{pdfcenterwindow}[]{}
\define@key{LWR@hyperref}{pdfdisplaydoctitle}[]{}
\define@key{LWR@hyperref}{pdfa}[]{}
\define@key{LWR@hyperref}{pdfnewwindow}[]{}
\define@key{LWR@hyperref}{pdflang}[]{}
\define@key{LWR@hyperref}{pdfpagelabels}[]{}
\define@key{LWR@hyperref}{pdfescapeform}[]{}
\define@key{LWR@hyperref}{english}[]{}
\define@key{LWR@hyperref}{UKenglish}[]{}
\define@key{LWR@hyperref}{british}[]{}
\define@key{LWR@hyperref}{USenglish}[]{}
\define@key{LWR@hyperref}{american}[]{}
\define@key{LWR@hyperref}{german}[]{}
\define@key{LWR@hyperref}{austrian}[]{}
\define@key{LWR@hyperref}{ngerman}[]{}
\define@key{LWR@hyperref}{naustrian}[]{}
\define@key{LWR@hyperref}{russian}[]{}
\define@key{LWR@hyperref}{brazil}[]{}
\define@key{LWR@hyperref}{brazilian}[]{}
\define@key{LWR@hyperref}{portuguese}[]{}
\define@key{LWR@hyperref}{spanish}[]{}
\define@key{LWR@hyperref}{catalan}[]{}
\define@key{LWR@hyperref}{afrikaans}[]{}
\define@key{LWR@hyperref}{french}[]{}
\define@key{LWR@hyperref}{frenchb}[]{}
\define@key{LWR@hyperref}{francais}[]{}
\define@key{LWR@hyperref}{acadian}[]{}
\define@key{LWR@hyperref}{canadien}[]{}
\define@key{LWR@hyperref}{italian}[]{}
\define@key{LWR@hyperref}{magyar}[]{}
\define@key{LWR@hyperref}{hungarian}[]{}
\define@key{LWR@hyperref}{greek}[]{}
\define@key{LWR@hyperref}{dutch}[]{}
```

```
\define@key{LWR@hyperref}{tex4ht}[]{}
\define@key{LWR@hyperref}{pdftex}[]{}
\define@key{LWR@hyperref}{luatex}[]{}
\define@key{LWR@hyperref}{nativepdf}[]{}
\define@key{LWR@hyperref}{dvipdfm}[]{}
\define@key{LWR@hyperref}{dvipdfmx}[]{}
\define@key{LWR@hyperref}{xetex}[]{}
\define@key{LWR@hyperref}{pdfmark}[]{}
\define@key{LWR@hyperref}{dvips}[]{}
\define@key{LWR@hyperref}{hypertex}[]{}
\define@key{LWR@hyperref}{vtex}[]{}
\define@key{LWR@hyperref}{vtexpdfmark}[]{}
\define@key{LWR@hyperref}{dviwindo}[]{}
\define@key{LWR@hyperref}{dvipsone}[]{}
\define@key{LWR@hyperref}{textures}[]{}
\define@key{LWR@hyperref}{latex2html}[]{}
\define@key{LWR@hyperref}{ps2pdf}[]{}
\define@key{LWR@hyperref}{vietnamese}[]{}
\define@key{LWR@hyperref}{vietnam}[]{}
\define@key{LWR@hyperref}{arabic}[]{}
\define@key{LWR@hyperref}{hidelinks}[]{}
\define@key{LWR@hyperref}{draft}[]{}
\define@key{LWR@hyperref}{nolinks}[]{}
\define@key{LWR@hyperref}{final}[]{}
\define@key{LWR@hyperref}{pdfa}[]{}
\define@key{LWR@hyperref}{pdfversion}[]{}
\define@key{LWR@hyperref}{typexml}[]{}
\define@key{LWR@hyperref}{tex4ht}[]{}
\define@key{LWR@hyperref}{pdftex}[]{}
\define@key{LWR@hyperref}{nativepdf}[]{}
\define@key{LWR@hyperref}{dvipdfm}[]{}
\define@key{LWR@hyperref}{dvipdfmx}[]{}
\define@key{LWR@hyperref}{dvipdfmx-outline-open}[]{}
\define@key{LWR@hyperref}{pdfmark}[]{}
\define@key{LWR@hyperref}{dvips}[]{}
\define@key{LWR@hyperref}{hypertex}[]{}
\define@key{LWR@hyperref}{vtex}[]{}
\define@key{LWR@hyperref}{vtexpdfmark}[]{}
\define@key{LWR@hyperref}{dviwindo}[]{}
\define@key{LWR@hyperref}{dvipsone}[]{}
\define@key{LWR@hyperref}{textures}[]{}
\define@key{LWR@hyperref}{latex2html}[]{}
\define@key{LWR@hyperref}{ps2pdf}[]{}
\define@key{LWR@hyperref}{xetex}[]{}
\define@key{LWR@hyperref}{driverfallback}[]{}
\define@key{LWR@hyperref}{customdriver}[]{}
\define@key{LWR@hyperref}{pdfversion}[]{}
\define@key{LWR@hyperref}{bookmarks}[]{}
\define@key{LWR@hyperref}{ocgcolorlinks}[]{}
\define@key{LWR@hyperref}{colorlinks}[]{}
\define@key{LWR@hyperref}{frenchlinks}[]{}
\define@key{LWR@hyperref}{backref}[]{}
\define@key{LWR@hyperref}{pagebackref}[]{}
\define@key{LWR@hyperref}{destlabel}[]{}
\define@key{LWR@hyperref}{pdfpagescrop}[]{}
\define@key{LWR@hyperref}{pdfpagemode}[]{}
\define@key{LWR@hyperref}{pdfnonfullscreenpagemode}[]{}
\define@key{LWR@hyperref}{pdfdirection}[]{}
\define@key{LWR@hyperref}{pdfviewarea}[]{}
\define@key{LWR@hyperref}{pdfviewclip}[]{}
```

```
\define@key{LWR@hyperref}{pdfprintarea}[]{}
\define@key{LWR@hyperref}{pdfprintclip}[]{}
\define@key{LWR@hyperref}{pdfprintscaling}[]{}
\define@key{LWR@hyperref}{pdfduplex}[]{}
\define@key{LWR@hyperref}{pdfpicktraybypdfsize}[]{}
\define@key{LWR@hyperref}{pdfprintpagerange}[]{}
\define@key{LWR@hyperref}{pdfnumcopies}[]{}
\define@key{LWR@hyperref}{pdfstartview}[]{}
\define@key{LWR@hyperref}{pdfstartpage}[]{}
\define@key{LWR@hyperref}{pdftoolbar}[]{}
\define@key{LWR@hyperref}{pdfmenubar}[]{}
\define@key{LWR@hyperref}{pdfwindowui}[]{}
\define@key{LWR@hyperref}{pdffitwindow}[]{}
\define@key{LWR@hyperref}{pdfcenterwindow}[]{}
\define@key{LWR@hyperref}{pdfdisplaydoctitle}[]{}
\define@key{LWR@hyperref}{pdfpagelayout}[]{}
\define@key{LWR@hyperref}{pdflang}[]{}
\define@key{LWR@hyperref}{baseurl}[]{}
\define@key{LWR@hyperref}{pdfusetitle}[]{}
\define@key{LWR@hyperref}{pdfpagelabels}[]{}
\define@key{LWR@hyperref}{hyperfootnotes}[]{}
\define@key{LWR@hyperref}{hyperfigures}[]{}
\define@key{LWR@hyperref}{hyperindex}[]{}
\define@key{LWR@hyperref}{encap}[]{}
\define@key{LWR@hyperref}{linkcolor}[]{}
\define@key{LWR@hyperref}{anchorcolor}[]{}
\define@key{LWR@hyperref}{citecolor}[]{}
\define@key{LWR@hyperref}{filecolor}[]{}
\define@key{LWR@hyperref}{urlcolor}[]{}
\define@key{LWR@hyperref}{menucolor}[]{}
\define@key{LWR@hyperref}{runcolor}[]{}
\define@key{LWR@hyperref}{allcolors}[]{}

\DeclareStringOption[false]{backref}[section]

\DeclareBoolOption{pagebackref}

\DeclareDefaultOption{}

\ProcessKeyvalOptions*\relax
```

Maybe load backref:

```
\ifdefstring{\LWR@hyperref@backref}{section}
    {\RequirePackage{backref}}
    {}

\ifdefstring{\LWR@hyperref@backref}{slide}
    {\RequirePackage{backref}}
    {}

\ifdefstring{\LWR@hyperref@backref}{page}
    {\RequirePackage{backref}}
    {}

\ifLWR@hyperref@pagebackref
    \RequirePackage{backref}
\fi
```

```
\LetLtxMacro\href\LWR@href
\LetLtxMacro\nolinkurl\LWR@nolinkurl
\LetLtxMacro\url\LWR@url
\LetLtxMacro\phantomsection\LWR@phantomsection

\newcommand*{\hyperbaseurl}[1]{}
```

No application for lwarp:

```
\newcommand*{\HyperDestNameFilter}[1]{#1}
\newcommand*{\HyperDestLabelReplace}[1]{#1}
\newcommand*{\HyperDestRename}[2]{}
```

No application for lwarp:

```
\newcommand*{\hyperget}[2]{}
```

\hyperimage {⟨*URL*⟩} {⟨*alt text*⟩}

Insert an image with alt text:

```
\NewDocumentCommand{\LWR@hyperimageb}{m +m}{%
    \LWR@ensuredoingapar%
    \def\LWR@templink{#1}%
    \@onelevel@sanitize\LWR@templink%
    \LWR@htmltag{%
        img src=\textquotedbl\LWR@templink\textquotedbl\ %
        alt=\textquotedbl#2\textquotedbl\ %
        class=\textquotedbl{}hyperimage\textquotedbl%
    }%
    \LWR@ensuredoingapar%
    \endgroup%
}

\newrobustcmd*{\hyperimage}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hyperimageb%
}

```

\hyperdef {⟨*1: category*⟩} {⟨*2: name*⟩} {⟨*3: text*⟩}

Creates an HTML anchor to `category.name` with the given text.

```
\NewDocumentCommand{\LWR@hyperdefb}{m m +m}{%
    \LWR@ensuredoingapar%
    \LWR@label@createtag{#1.#2}%
    #3%
    \endgroup%
}

\newcommand*{\hyperdef}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hyperdefb%
}

```

\LWR@hyperrefb {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩}

Creates an HTML link to URL#category.name with the given text.

```
\newcommand{\LWR@hyperreffinish}[1]{%
    \begingroup%
    \RenewDocumentCommand{\ref}{s m}{\LWR@print@ref{##2}}%
    #1%
    \endgroup%
    \LWR@htmltag{/a}%
}

\newcommand*{\LWR@hyperrefbb}[3]{%
    \LWR@htmltag{%
        a href=\textquotedbl%
            \detokenize\expandafter{#1}\LWR@hashmark%
            \detokenize\expandafter{#2}.\detokenize\expandafter{#3}%
        \textquotedbl%
        \LWR@addlinktitle%
    }%
    \endgroup%
    \LWR@hyperreffinish%
}

\newrobustcmd*{\LWR@hyperrefb}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hyperrefbb%
}
```

\LWR@hyperrefc [⟨*label*⟩] {⟨*text*⟩}

Creates text as an HTML link to the LaTeX label.

```
\NewDocumentCommand{\LWR@hyperrefcb}{O{label}}{%
    \LWR@startref{#1}%
    \endgroup%
    \LWR@hyperreffinish%
}

\newcommand*{\LWR@hyperrefc}{%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hyperrefcb%
}
```

\hyperref {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩} — or — [⟨*1: label*⟩] {⟨*2: text*⟩}

```
\DeclareRobustCommand*{\hyperref}{%
    \LWR@ensuredoingapar%
    \@ifnextchar[\LWR@hyperrefc\LWR@hyperrefb%
}
```

\hypertarget {⟨*name*⟩} {⟨*text*⟩}

Creates an anchor to name with the given text.

```
\NewDocumentCommand{\LWR@hypertargetb}{m +m}{%
    \label{LWR-ht-#1}%
    #2%
```

```
    \endgroup%
}

\newcommand*{\hypertarget}{%
    \LWR@ensuredoingapar%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hypertargetb%
}
```

\hyperlink {⟨*name*⟩} {⟨*text*⟩}

Creates a link to the anchor created by hypertarget, with the given link text.

Declared because also defined by memoir.

```
\DeclareDocumentCommand{\LWR@hyperlinkb}{m}{%
    \ifbool{LWR@insidemathcomment}%
        {\endgroup}%
        {\LWR@hyperrefcb[LWR-ht-#1]}%
}

\DeclareDocumentCommand{\hyperlink}{}{%
    \LWR@ensuredoingapar%
    \begingroup%
    \LWR@linkcatcodes%
    \LWR@hyperlinkb%
}
```

\LWR@nullify@hyperref {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩} — or — [⟨*1: label*⟩] {⟨*2: text*⟩}

```
\newcommand{\LWR@nullify@hyperrefb}[2][]{}

\newcommand*{\LWR@nullify@hyperref}{%
    \@ifnextchar[\LWR@nullify@hyperrefb\@fourthoffour%
}
```

To nullify in a lateximage or SVG math. \hypertarget must be left active for references to work, and does not harm.

```
\appto\LWR@restoreorigformatting{%
    \LetLtxMacro\hyperdef\@thirdofthree
    \LetLtxMacro\hyperlink\@secondoftwo%
    \LetLtxMacro\hyperref\LWR@nullify@hyperref%
}
```

\autoref * {⟨*label*⟩}

For HTML, \cleveref is used instead.

```
\NewDocumentCommand{\autoref}{s m}{%
    \IfBooleanTF{#1}{\ref{#2}}{\cref{#2}}%
}
```

\autopageref {⟨*label*⟩}

For HTML, \cleveref is used instead.

```
\NewDocumentCommand{\autopageref}{s m}{%
```

```
    \IfBooleanTF{#1}{\cpageref{#2}}{\cref{#2}}%
}
```

Default names:

```
\def\equationautorefname{Equation}%
\def\footnoteautorefname{footnote}%
\def\itemautorefname{item}%
\def\figureautorefname{Figure}%
\def\tableautorefname{Table}%
\def\partautorefname{Part}%
\def\appendixautorefname{Appendix}%
\def\chapterautorefname{chapter}%
\def\sectionautorefname{section}%
\def\subsectionautorefname{subsection}%
\def\subsubsectionautorefname{subsubsection}%
\def\paragraphautorefname{paragraph}%
\def\subparagraphautorefname{subparagraph}%
\def\FancyVerbLineautorefname{line}%
\def\theoremautorefname{Theorem}%
\def\pageautorefname{page}%
```

\pdfstringdef {⟨*macroname*⟩} {⟨*TEXstring*⟩}

```
\newcommand{\pdfstringdef}[2]{}
```

\pdfbookmark [⟨*level*⟩] {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\pdfbookmark}[3][]{}
```

\currentpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\currentpdfbookmark}[2]{}
```

\subpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\subpdfbookmark}[2]{}
```

\belowpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\belowpdfbookmark}[2]{}
```

\texorpdfstring {⟨*TEXstring*⟩} {⟨*PDFstring*⟩}

```
\let\texorpdfstring\relax
\newcommand{\texorpdfstring}[2]{#1}
```

{⟨*commands*⟩}

\pdfstringdefDisableCommands

```
\newcommand{\pdfstringdefDisableCommands}[1]{}
```

\hypercalcbp {⟨*dimen*⟩} From hyperref.

```
\def\hypercalcbp#1{%
    \strip@pt\dimexpr 0.99626401\dimexpr(#1)\relax\relax
}%
```

\Acrobatmenu {⟨*menuoption*⟩} {⟨*text*⟩}

```
402 \newcommand{\Acrobatmenu}[2]{}
```

\TextField [⟨*parameters*⟩] {⟨*label*⟩}

```
403 \DeclareRobustCommand{\TextField}[2][]{}
```

\CheckBox [⟨*parameters*⟩] {⟨*label*⟩}

```
404 \DeclareRobustCommand{\CheckBox}[2][]{}
```

\ChoiceMenu [⟨*parameters*⟩] {⟨*label*⟩} {⟨*choices*⟩}

```
405 \DeclareRobustCommand{\ChoiceMenu}[3][]{}
```

\PushButton [⟨*parameters*⟩] {⟨*label*⟩}

```
406 \DeclareRobustCommand{\PushButton}[2][]{}
```

\Submit [⟨*parameters*⟩] {⟨*label*⟩}

```
407 \DeclareRobustCommand{\Submit}[2][]{}
```

\Reset [⟨*parameters*⟩] {⟨*label*⟩}

```
408 \DeclareRobustCommand{\Reset}[2][]{}
```

\Gauge [⟨*parameters*⟩] {⟨*label*⟩}

```
409 \DeclareRobustCommand{\Gauge}[2][]{}
```

\LayoutTextField {⟨*label*⟩} {⟨*field*⟩}

```
410 \newcommand*{\LayoutTextField}[2]{}
```

\LayoutChoiceField {⟨*label*⟩} {⟨*field*⟩}

```
411 \newcommand*{\LayoutChoiceField}[2]{}
```

\LayoutCheckField {⟨*label*⟩} {⟨*field*⟩}

```
412 \newcommand*{\LayoutCheckField}[2]{}
```

\MakeRadioField {⟨*width*⟩} {⟨*height*⟩}

```
413 \newcommand*{\MakeRadioField}[2]{}
```

\MakeCheckField {⟨*width*⟩} {⟨*height*⟩}

```
414 \newcommand*{\MakeCheckField}[2]{}
```

\MakeTextField {⟨*width*⟩} {⟨*height*⟩}

```
\newcommand*{\MakeTextField}[2]{}
```

\MakeChoiceField {⟨*width*⟩} {⟨*height*⟩}

```
\newcommand*{\MakeChoiceField}[2]{}
```

\MakeFieldButton {⟨*text*⟩}

```
\newcommand{\MakeFieldButton}[1]{}
```

File 225 **lwarp-hyperxmp.sty**

§ 334 Package

hyperxmp

hyperxmp (*Pkg*) hyperxmp is ignored.

for HTML output: Discard all options for lwarp-hyperxmp:

```
\LWR@ProvidesPackageDrop{hyperxmp}[2018/11/27]

\define@key{LWR@hyperref}{pdfdate}[]{}
\define@key{LWR@hyperref}{pdfmetadate}[]{}
\define@key{LWR@hyperref}{pdfcopyright}[]{}
\define@key{LWR@hyperref}{pdftype}[]{}
\define@key{LWR@hyperref}{pdflicenseurl}[]{}
\define@key{LWR@hyperref}{pdfauthortitle}[]{}
\define@key{LWR@hyperref}{pdfcaptionwriter}[]{}
\define@key{LWR@hyperref}{pdfmetalang}[]{}
\define@key{LWR@hyperref}{pdfapart}[]{}
\define@key{LWR@hyperref}{pdfaconformance}[]{}
\define@key{LWR@hyperref}{pdfuapart}[]{}
\define@key{LWR@hyperref}{pdfxstandard}[]{}
\define@key{LWR@hyperref}{pdfsource}[]{}
\define@key{LWR@hyperref}{pdfdocumentid}[]{}
\define@key{LWR@hyperref}{pdfinstanceid}[]{}
\define@key{LWR@hyperref}{pdfversionid}[]{}
\define@key{LWR@hyperref}{pdfrendition}[]{}
\define@key{LWR@hyperref}{pdfpublication}[]{}
\define@key{LWR@hyperref}{pdfpubtype}[]{}
\define@key{LWR@hyperref}{pdfbytes}[]{}
\define@key{LWR@hyperref}{pdfnumpages}[]{}
\define@key{LWR@hyperref}{pdfissn}[]{}
\define@key{LWR@hyperref}{pdfeissn}[]{}
\define@key{LWR@hyperref}{pdfisbn}[]{}
\define@key{LWR@hyperref}{pdfbookedition}[]{}
\define@key{LWR@hyperref}{pdfpublisher}[]{}
\define@key{LWR@hyperref}{pdfvolumenum}[]{}
\define@key{LWR@hyperref}{pdfissuenum}[]{}
\define@key{LWR@hyperref}{pdfpagerange}[]{}
\define@key{LWR@hyperref}{pdfdoi}[]{}
\define@key{LWR@hyperref}{pdfurl}[]{}
\define@key{LWR@hyperref}{pdfidentifier}[]{}
\define@key{LWR@hyperref}{pdfsubtitle}[]{}
\define@key{LWR@hyperref}{pdfpubstatus}[]{}
\define@key{LWR@hyperref}{pdfcontactaddress}[]{}
\define@key{LWR@hyperref}{pdfcontactcity}[]{}
\define@key{LWR@hyperref}{pdfcontactregion}[]{}
```

```
\define@key{LWR@hyperref}{pdfcontactpostcode}[]{}
\define@key{LWR@hyperref}{pdfcontactcountry}[]{}
\define@key{LWR@hyperref}{pdfcontactphone}[]{}
\define@key{LWR@hyperref}{pdfcontactemail}[]{}
\define@key{LWR@hyperref}{pdfcontacturl}[]{}
\define@key{LWR@hyperref}{keeppdfinfo}[]{}
\define@key{LWR@hyperref}{pdfauthor}[]{}
\define@key{LWR@hyperref}{pdfkeywords}[]{}
```

File 226 **lwarp-hyphenat.sty**

§ 335 Package hyphenat

hyphenat (*Pkg*) hyphenat is emulated during HTML output, while the print-mode version is used inside a `lateximage`.

for HTML output:

```
\LWR@ProvidesPackagePass{hyphenat}[2009/09/02]

\LetLtxMacro\LWRHYNAT@origtextnhtt\textnhtt
\LetLtxMacro\LWRHYNAT@orignhttfamily\nhttfamily
\LetLtxMacro\LWRHYNAT@orignohyphens\nohyphens
\LetLtxMacro\LWRHYNAT@origbshyp\bshyp
\LetLtxMacro\LWRHYNAT@origfshyp\fshyp
\LetLtxMacro\LWRHYNAT@origdothyp\dothyp
\LetLtxMacro\LWRHYNAT@origcolonhyp\colonhyp
\LetLtxMacro\LWRHYNAT@orighyp\hyp

\LetLtxMacro\textnhtt\texttt
\LetLtxMacro\nhttfamily\ttfamily

\renewcommand{\nohyphens}[1]{#1}
\renewrobustcmd{\bshyp}{%
    \ifmmode\backslash\else\textbackslash\fi%
}
\renewrobustcmd{\fshyp}{/}
\renewrobustcmd{\dothyp}{.}
\renewrobustcmd{\colonhyp}{:}
\renewrobustcmd{\hyp}{-}

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\textnhtt\LWRHYNAT@origtextnhtt%
\LetLtxMacro\nhttfamily\LWRHYNAT@orignhttfamily%
\LetLtxMacro\nohyphens\LWRHYNAT@orignohyphens%
\LetLtxMacro\bshyp\LWRHYNAT@origbshyp%
\LetLtxMacro\fshyp\LWRHYNAT@origfshyp%
\LetLtxMacro\dothyp\LWRHYNAT@origdothyp%
\LetLtxMacro\colonhyp\LWRHYNAT@origcolonhyp%
\LetLtxMacro\hyp\LWRHYNAT@orighyp%
}
```

File 227 **lwarp-idxlayout.sty**

§ 336 Package idxlayout

(*Emulates or patches code by* Thomas Titz.)

idxlayout (*Pkg*) idxlayout is emulated.

for HTML output: Discard all options for lwarp-idxlayout:

```
\LWR@ProvidesPackageDrop{idxlayout}[2012/03/30]

\newcommand{\LWR@indexprenote}{}
```

\AtBeginDocument to help with package load order.

```
\AtBeginDocument{
    \preto\printindex{

    \LWR@maybe@orignewpage
    \LWR@startpars

    \LWR@indexprenote

    }
}

\newcommand{\setindexprenote}[1]{\renewcommand{\LWR@indexprenote}{#1}}
\newcommand*{\noindexprenote}{\renewcommand{\LWR@indexprenote}{}}

\newcommand{\idxlayout}[1]{}
\newcommand*{\indexfont}{}
\newcommand*{\indexjustific}{}
\newcommand*{\indexsubsdelim}{}
\newcommand*{\indexstheadcase}{}
```

File 228 **lwarp-ifoddpage.sty**

§ 337 Package ifoddpage

(*Emulates or patches code by* Martin Scharrer.)

ifoddpage (*Pkg*) ifoddpage is emulated.

for HTML output: Discard all options for lwarp-ifoddpage:

```
\LWR@ProvidesPackageDrop{ifoddpage}[2016/04/23]

\newif\ifoddpage

\newif\ifoddpageoroneside

\DeclareRobustCommand{\checkoddpage}{\oddpagetrue\oddpageoronesidetrue}

\def\oddpage@page{1}

\def\@ifoddpage{%
        \expandafter\@firstoftwo
}

\def\@ifoddpageoroneside{%
        \expandafter\@firstoftwo
}
```

File 229 **lwarp-imakeidx.sty**

§ 338 Package

imakeidx

(*Emulates or patches code by* Enrico Gregorio.)

imakeidx (*Pkg*) imakeidx is patched for use by lwarp.

letter headings When using *makeindex*, to match the print and HTML output's display of index letter headings, specify the `lwarp.ist` style:

```
\makeindex[options={-s lwarp.ist}]
```

(For HTML the `lwarp.ist` style is used automatically, which displays letter headings. When using *xindy* the default style also displays letter headings.)

index setup See section 8.6.18 for how to setup *lwarpmk* to process the indexes with imakeidx, both with and without shell escape.

for HTML output:

```
\LWR@ProvidesPackagePass{imakeidx}[2016/10/15]
```

Use the new HTML suffix:

```
\catcode`\_=12%
\define@key{imki}{name}{\def\imki@name{#1_html}}
\catcode`\_=8%
```

\printindex The HTML version of \printindex:

```
\catcode`\_=12%

\renewcommand*{\printindex}[1][\imki@jobname]{%
\LWR@maybe@orignewpage%
\LWR@startpars%
\ifstrequal{#1}{\imki@jobname}{%
  \@ifundefined{#1@idxfile}{%
        \imki@error{#1}%
    }{%
        \imki@putindex{#1}%
    }%
}{%
 \@ifundefined{#1_html@idxfile}{\imki@error{#1_html}}{\imki@putindex{#1_html}}%
}%
}

\catcode`\_=8%
```

\@index The HTML version of \@index:

```
\catcode`\_=12%

\def\@index[#1]{%
    \ifstrequal{#1}{\imki@jobname}%
    {%
        \@ifundefined{#1@idxfile}%
```

```
        {%
            \PackageWarning{lwarp-imakeidx}{Undefined index file ‘#1’}%
            \begingroup
            \@sanitize
            \imki@nowrindex%
        }%
        {%
            \edef\@idxfile{#1}%
            \begingroup
            \@sanitize
            \@wrindex\@idxfile%
        }%
    }%
    {%
        \@ifundefined{#1_html@idxfile}%
        {%
            \PackageWarning{lwarp-imakeidx}{Undefined index file ‘#1_html’}%
            \begingroup
            \@sanitize
            \imki@nowrindex%
        }%
        {%
            \edef\@idxfile{#1_html}%
            \begingroup
            \@sanitize
            \@wrindex\@idxfile%
        }%
    }%
}

\catcode‘\_=8%
```

\item
\subitem
\subsubitem

HTML versions of \item, etc.:

```
\appto\theindex{%
    \LetLtxMacro\item\LWR@indexitem%
    \LetLtxMacro\subitem\LWR@indexsubitem%
    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
}
```

\imki@wrindexentrysplit {⟨*file*⟩} {⟨*entry*⟩} {⟨*page*⟩}

\imki@wrindexentryunique {⟨*file*⟩} {⟨*entry*⟩} {⟨*page*⟩}

While writing index entries, adds an HTML label, and writes the label's index instead of the page number:

```
\renewcommand\imki@wrindexentrysplit[3]{%
    \addtocounter{LWR@autoindex}{1}%
    \expandafter\protected@write\csname#1@idxfile\endcsname{}%
        {\string\indexentry{#2}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
    \label{LWRindex-\arabic{LWR@autoindex}}%
}

\renewcommand\imki@wrindexentryunique[3]{%
    \addtocounter{LWR@autoindex}{1}%
```

```
    \protected@write\@indexfile{}%
        {\string\indexentry[#1]{#2}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
    \label{LWRindex-\arabic{LWR@autoindex}}%
}

\def\imki@wrindexsplit#1#2{%
\imki@wrindexentrysplit{#1}{#2}{\thepage}%
\endgroup\imki@showidxentry{#1}{#2}%
  \@esphack%
}

\def\imki@wrindexunique#1#2{%
\imki@wrindexentryunique{#1}{#2}{\thepage}%
\endgroup\imki@showidxentry{#1}{#2}%
  \@esphack%
  }

```

`\LWR@imki@setxdydefopts`

Sets the *xindy* HTML options, ignoring the user's settings.

```
\newcommand*{\LWR@imki@setxdydefopts}{%
    \edef\imki@options{ \space %
        -M \space \LWR@xindyStyle\space %
        -L \space \LWR@xindyLanguage\space %
        -C \space \LWR@xindyCodepage\space %
    }%
}
```

`\LWR@imki@setdefopts` {⟨*user options*⟩}

Sets the HTML options, added to the user's settings, depending on whether *makeindex* or *xindy* are used.

For *makeindex*, the user's choice is ignored, and only the lwarp version is used. (Only one style at a time is possible.)

For *xindy*, multiple modules may be specified, and the lwarp version is appended.

```
\newcommand*{\LWR@imki@setdefopts}[1]{%
\ifblank{#1}{%
    \edef\imki@options{\space -s \space \LWR@makeindexStyle \space}%
    \ifdefstring{\imki@progdefault}{xindy}{\LWR@imki@setxdydefopts}{}%
    \ifdefstring{\imki@progdefault}{texindy}{\LWR@imki@setxdydefopts}{}%
    \ifdefstring{\imki@progdefault}{truexindy}{\LWR@imki@setxdydefopts}{}%
}{%
    \edef\imki@options{\space #1 \space}%
}%
}
```

`\imki@makeindex`

Use the new HTML options:

```
\xpatchcmd{\imki@makeindex}
    {\let\imki@options\space}
    {\LWR@imki@setdefopts{}}%
    {}
    {\LWR@patcherror{imakeidx}{makeindex}}
```

Use the new HTML options.

```
\define@key{imki}{options}{\LWR@imki@setdefopts{#1}}
```

\imki@resetdefaults

Use the new HTML options:

```
\xpatchcmd{\imki@resetdefaults}
    {\def\imki@options{ }}
    {\LWR@imki@setdefopts{}}
    {}
    {\LWR@patcherror{imakeidx}{resetdefaults}}
```

theindex was already defined \AtBeginDocument by the lwarp core, so it must be redefined here similarly, but patched for imakeidx:

Env theindex

```
\AtBeginDocument{
\renewenvironment*{theindex}{%
    \imki@maybeaddtotoc
     \imki@indexlevel{\indexname}
    \LetLtxMacro\item\LWR@indexitem%
    \LetLtxMacro\subitem\LWR@indexsubitem%
    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
}{}
}% AtBeginDocument
```

Update to the new defaults:

```
\imki@resetdefaults
```

Update to the new patches:

\AtBeginDocument is because \@wrindex is previously defined as \AtBeginDocument in the lwarp core.

```
\ifimki@splitindex
  \let\imki@startidx\imki@startidxunique
  \AtBeginDocument{\let\@wrindex\imki@wrindexunique}
  \let\imki@putindex\imki@putindexunique
  \let\imki@wrindexentry\imki@wrindexentryunique
  \let\imki@startidxsplit\@undefined
  \let\imki@wrindexsplit\@undefined
  \let\imki@putindexsplit\@undefined
\else
  \let\imki@startidx\imki@startidxsplit
  \AtBeginDocument{\let\@wrindex\imki@wrindexsplit}
  \let\imki@putindex\imki@putindexsplit
  \let\imki@wrindexentry\imki@wrindexentrysplit
  \let\imki@startidxunique\@undefined
  \let\imki@wrindexunique\@undefined
  \let\imki@putindexunique\@undefined
\fi
```

File 230 **lwarp-impnattypo.sty**

§ 339 Package impnattypo

impnattypo (*Pkg*) impnattypo is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{impnattypo}[2019/03/04]
```

File 231 **lwarp-index.sty**

§ 340 Package index

(*Emulates or patches code by* David M. Jones.)

index (*Pkg*) index is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{index}[2004/01/20]
```

Use \theLWR@autoindex instead of \thepage. \@tempswatrue is used to force an immediate write to the index file instead of waiting until the end of the page.

```
\xpatchcmd{\newindex}
    {\x@newindex[thepage]}
    {%
        \@tempswatrue%
        \x@newindex[theLWR@autoindex]%
    }
    {}
    {\LWR@patcherror{index}{newindex}}

\xpatchcmd{\renewindex}
    {\x@renewindex[thepage]}
    {%
        \@tempswatrue%
        \x@renewindex[theLWR@autoindex]%
    }
    {}
    {\LWR@patcherror{index}{renewindex}}
```

Patched to set a new autoindex:

```
\xpatchcmd{\@wrindex}
    {\begingroup}
    {%
        \addtocounter{LWR@autoindex}{1}%                    lwarp
        \label{LWRindex-\arabic{LWR@autoindex}}%    lwarp
        \begingroup%
    }
    {}
    {\LWR@patcherror{index}{@wrindex}}
```

`\AtBeginDocument` lwarp core `\lets` `\@wrindex` to `\LWR@wrindex`. Since the index package has been loaded, `\let` to its version instead:

```
\let\LWR@index@wrindex\@wrindex

\AtBeginDocument{
\let\@wrindex\LWR@index@wrindex
}
```

Modified to add `\index@prologue`:

```
\AtBeginDocument{
\renewenvironment*{theindex}{%
    \LWR@indexsection{\indexname}%
    \ifx\index@prologue\@empty\else
        \index@prologue
        \bigskip
    \fi
    \LetLtxMacro\item\LWR@indexitem%
    \LetLtxMacro\subitem\LWR@indexsubitem%
    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
}{}
}% AtBeginDocument
```

Disabled:

```
\def\@showidx#1{}
\let\@texttop\relax
\renewcommand*{\raggedbottom}{}
\renewcommand*{\flushbottom}{}
\renewcommand*{\markboth}[2]{}
\renewcommand*{\markright}[1]{}
```

File 232 **lwarp-inputtrc.sty**

§ 341 Package

inputtrc

(*Emulates or patches code by* Uwe Lück.)

`inputtrc` (*Pkg*) inputtrc is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{inputtrc}[2012/10/10]
```

Patched to remove extraneous spaces, which sometimes showed up in logos inside a `lateximage`.

```
\renewcommand*{\IT@prim@input}[1]{%
  \typeout{\IT@indent\IT@currfile INPUTTING #1}%
%% ... TODO: option to write to '.log' only.
  \xdef\IT@filestack{{\IT@currfile}\IT@filestack}%
  \xdef\IT@currfile{#1}%
  \expandafter \gdef\expandafter \IT@indent\expandafter{%
    \IT@indent \IT@indent@unit}%               lwarp
  \@@input#1%                                  lwarp
  \expandafter\IT@pop@indent\IT@indent   \@nil% lwarp
  \expandafter\IT@pop@file  \IT@filestack\@nil% lwarp
```

```
12   \IT@maybe@returnmessage%% v0.2                lwarp
13 }
```

File 233 **lwarp-intopdf.sty**

§ 342 Package intopdf

intopdf (*Pkg*) intopdf is emulated.

The filespec, MIME type, and description are ignored for now.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{intopdf}[2019/05/28]

2 \NewDocumentCommand{\attachandlink}{o m o m m}{%
3     \LWR@href{#2}{#5}%
4 }
```

File 234 **lwarp-isomath.sty**

§ 343 Package isomath

(*Emulates or patches code by* Günter Milde.)

isomath (*Pkg*) isomath is used as-is for SVG math, and emulated for MathJax.

⚠ MathJax sans MathJax does not provide a sans math font, so sans is typeset as roman.

for HTML output:

```
1 \LWR@ProvidesPackagePass{isomath}[2012/09/04]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\let\mathbfit\boldsymbol}
4 \CustomizeMathJax{\let\mathsfbfit\mathbfit}% not sans
5 \CustomizeMathJax{\let\mathsfit\mathit}% not sans
6 \CustomizeMathJax{\let\vectorsym\mathbfit}
7 \CustomizeMathJax{\let\matrixsym\mathbfit}
8 \CustomizeMathJax{\let\tensorsym\mathsfbfit}
9 \CustomizeMathJax{\let\mathboldsans\mathsfbfit}
10 \CustomizeMathJax{\let\mathbold\mathbfit}
11 \CustomizeMathJax{\let\mathsans\mathrm}% not sans
12 \end{warpMathJax}
```

File 235 **lwarp-isotope.sty**

§ 344 Package isotope

(*Emulates or patches code by* Heiko Bauke.)

isotope (*Pkg*) isotope is patched for use by lwarp with SVG math, and emulated for MathJax.

for HTML output:

```
1 \LWR@ProvidesPackagePass{isotope}[2011/08/26]
```

```
\newcommand{\LWR@HTML@isotope@two}[2][]{%
    \renewcommand{\isotope@atomicnumber}{#1}%
    \edef\LWR@isotope@alttag{%
        \textbackslash(
        \textbackslash{}isotope
        [\isotope@nucleonnumber]%
        [\isotope@atomicnumber]%
        \{#2\}
        \textbackslash)%
    }%
  \ifbool{mathjax}%
    {\LWR@isotope@alttag}%
    {% SVG
        \m@th%
        \LWR@subsingledollar*%
        {% alt tag
            \LWR@isotope@alttag%
        }%
        {isotope}% add'l hashing
        {% contents
            \settowidth\@tempdimb{%
                \ensuremath{\scriptstyle\isotope@nucleonnumber}%
            }%
            \settowidth\@tempdimc{%
                \ensuremath{\scriptstyle\isotope@atomicnumber}%
            }%
            \ifdim\@tempdimb<\@tempdimc\@tempdimb=\@tempdimc\fi%
            \ensuremath{
                {}%
                ^{\makebox[\@tempdimb][r]{%
                    \ensuremath{%
                    \scriptstyle\isotope@nucleonnumber%
                    }% ensuremath
                }}%
                _{\makebox[\@tempdimb][r]{%
                    \ensuremath{%
                        \scriptstyle\isotope@atomicnumber%
                    }% ensuremath
                }}%
                \isotopestyle{#2}%
            }% ensuremath
        }% contents
    }% SVG
  \endgroup%
}%
\LWR@formatted{isotope@two}

\begin{warpMathJax}
\CustomizeMathJax{%
    \newcommand{\LWRisotopetwo}[2][]{%
        {%
            \vphantom{\mathrm{#2}}%
            {}^{\LWRisotopenucleonnumber}_{#1}%
            \mathrm{#2}%
        }%
    }%
}

\CustomizeMathJax{%
    \newcommand{\isotope}[1][]{%
```

```
        \def\LWRisotopenucleonnumber{#1}%
        \LWRisotopetwo%
    }%
}
\end{warpMathJax}
```

File 236 **lwarp-jurabib.sty**

§ 345 Package jurabib

(*Emulates or patches code by* Jens Berger.)

jurabib (*Pkg*) jurabib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{jurabib}[2004/01/25]

\renewrobustcmd{\jblangle}{\textless}

\renewrobustcmd{\jbrangle}{\textgreater}

\renewcommand*{\jb@biblaw@item}{%
    \hspace{0.5em}%
%     $\triangleright$
    \HTMLunicode{25B7}%     lwarp%
    \hspace{0.5em}%
}

\renewrobustcmd{\jbarchsig}[2]{%
     \ifjbweareinbib
        \settowidth{\jb@subarchitemwidth}{\jbsamesubarchindent+#1}%
     \setlength{\jb@subarchentrywidth}{\textwidth-\jb@subarchitemwidth-4em}%
%      \begin{tabular}{@{}p{\jb@subarchitemwidth}@{}j{\jb@subarchentrywidth}@{}}%
           #1\ifjb@dot\unskip\unskip\unskip.\fi
%             &
            \quad%      lwarp
            \ifthenelse{\equal{#2}{}}{}{\jbarchnameformat{#2}}%
%         \end{tabular}
     \fi
}%


\xpatchcmd{\jb@do@post@item}
    {\begin{tabular}{p{\jb@biblaw@item@width}j{\jb@biblaw@entry@width}}}
    {}
    {}
    {\LWR@patcherror{jurabib}{jb@do@post@item 1}}

\xpatchcmd{\jb@do@post@item}
    {\multicolumn{2}{p{\columnwidth}}{\jb@@name}}
    {\jb@@name}
    {}
    {\LWR@patcherror{jurabib}{jb@do@post@item 2}}

\xpatchcmd{\jb@do@post@item}
    {\jb@biblaw@item & \jb@@fulltitle}
    {\jb@biblaw@item \quad \jb@@fulltitle}
    {}
```

```
    {\LWR@patcherror{jurabib}{jb@do@post@item 3}}

\xpatchcmd{\jb@do@post@item}
    {\end{tabular}}
    {}
    {}
    {\LWR@patcherror{jurabib}{jb@do@post@item 4}}

\xpatchcmd{\jb@do@post@item}
    {\begin{minipage}[t]{\bibnumberwidth}}
    {}
    {}
    {\LWR@patcherror{jurabib}{jb@do@post@item 5}}

\xpatchcmd{\jb@do@post@item}
    {\end{minipage}}
    {\quad}
    {}
    {\LWR@patcherror{jurabib}{jb@do@post@item 6}}
```

File 237 **lwarp-karnaugh-map.sty**

§ 346 Package

karnaugh-map

(*Emulates or patches code by* Mattias Jacobsson.)

karnaugh-map (*Pkg*) karnaugh-map is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{karnaugh-map}[2017/02/20]
```

This patch is needed only because lwarp changes the definition of \&, and the original uses \ifnum to compare 0 with \&. It is hard to patch this environment, so the entire thing is redefined here, with the lwarp modifications identified in comments.

```
\RenewDocumentEnvironment{karnaugh-map}{s O{4} O{4} O{1} O{$X_1X_0$} O{$X_3X_2$} O{$X_5X_4$}} {%
  \begingroup
    % store map size {[START]
      \renewcommand{\@karnaughmap@var@mapsizex@}{#2}%
      \renewcommand{\@karnaughmap@var@mapsizey@}{#3}%
      \renewcommand{\@karnaughmap@var@mapsizez@}{#4}%
    % [END]}
    % determinate if markings should be color or black and white
    \IfBooleanTF{#1}{%
      % should be black and white
      \renewcommand{\@karnaughmap@var@bw@}{1}%
    }{%
      % should be color
      \renewcommand{\@karnaughmap@var@bw@}{0}%
    }%
    %
    % find matching matrix template and alignment parameters {[START]
    \newcommand{\@karnaughmap@local@matrixtemplate@}{0}% '0' is considered as missing matrix template
      \newcommand{\@karnaughmap@local@maprealignmentx@}{0}%
      \newcommand{\@karnaughmap@local@maprealignmenty@}{0}%
    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=221
        \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
```

```
                \&                 0 \&                  1 \& \phantom{0} \\
             0 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \&         \\
             1 \& |(000010)| \phantom{0} \& |(000011)| \phantom{0} \&         \\
     \phantom{0}  \&                   \&                   \&          \\
       }%
     \fi
   \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=241
       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
                \&                 0 \&                  1 \& \phantom{00} \\
           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \&          \\
           01 \& |(000010)| \phantom{0} \& |(000011)| \phantom{0} \&          \\
           11 \& |(000110)| \phantom{0} \& |(000111)| \phantom{0} \&          \\
           10 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \&          \\
     \phantom{00} \&                   \&                   \&          \\
       }%
     \fi
   \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=421
       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
               \&                00 \&                01 \&                11 \&                10 \& \phan
           0  \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(000
           1  \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(000
     \phantom{00} \&                   \&                   \&                   \&                   \&
       }%
     \fi
   \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=441
       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
               \&                00 \&                01 \&                11 \&                10 \& \phan
           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(000
           01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(000
           11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(000
           10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(000
     \phantom{00} \&                   \&                   \&                   \&                   \&
       }%
     \fi
   \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=442
       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
               \&                00 \&                01 \&                11 \&                10 \& \phan
           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(000
           01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(000
           11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(000
           10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(000
     \phantom{00} \&                   \&                   \&                   \&                   \&
       }%
       \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
     \fi
   \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=444
       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
               \&                00 \&                01 \&                11 \&                10 \& \phan
           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(000
           01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(000
           11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(000
           10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(000
     \phantom{00} \&                   \&                   \&                   \&                   \&
           00 \& |(100000)| \phantom{0} \& |(100001)| \phantom{0} \& |(100011)| \phantom{0} \& |(10
           01 \& |(100100)| \phantom{0} \& |(100101)| \phantom{0} \& |(100111)| \phantom{0} \& |(10
           11 \& |(101100)| \phantom{0} \& |(101101)| \phantom{0} \& |(101111)| \phantom{0} \& |(10
           10 \& |(101000)| \phantom{0} \& |(101001)| \phantom{0} \& |(101011)| \phantom{0} \& |(10
     \phantom{00} \&                   \&                   \&                   \&                   \&
       }%
       \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
```

```
84        \renewcommand{\@karnaughmap@local@maprealignmenty@}{-2.5}%
85      \fi
86    % [END]}
87   % test if a matrix template is found or not(aka "\@karnaughmap@local@matrixtemplate@" equals to '0')
88    \ifdefstring{\@karnaughmap@local@matrixtemplate@}{0}{%  lwarp
89 %     \ifnum0=\@karnaughmap@local@matrixtemplate@%  original
90      % print error if no template could be found
91      \PackageError{lwarp-karnaugh-map}{%
92        Can not find a template fitting your specification
93       (\@karnaughmap@var@mapsizex@\space x \@karnaughmap@var@mapsizey@\space x
94        \@karnaughmap@var@mapsizez@)%
95      }{%
96        Existing templates have the following dimensions:
97        2x2x1, 2x4x1, 4x2x1, 4x4x1, 4x4x2, and 4x4x4.
98      }%
99 %     \fi   original
100   }{\relax}%    lwarp
101   \begin{tikzpicture}
102     % grid
103     % for all dimensions
104   \draw[color=black, ultra thin] (0,0) grid (\@karnaughmap@var@mapsizex@,\@karnaughmap@var@mapsizey
105     % when there are 2 sub maps
106     \ifnum\@karnaughmap@var@mapsizez@=2
107       \draw[color=black, ultra thin] (5,0) grid (9,4);
108     \fi
109     % when there are 4 sub maps
110     \ifnum\@karnaughmap@var@mapsizez@=4
111       \draw[color=black, ultra thin] (5,0) grid (9,4);
112       \draw[color=black, ultra thin] (0,-5) grid (4,-1);
113       \draw[color=black, ultra thin] (5,-5) grid (9,-1);
114     \fi
115     % labels
116     % for all dimensions
117   \node[above] at (\@karnaughmap@var@mapsizex@*0.5,\@karnaughmap@var@mapsizey@+0.9) {\small{#5}};
118     \node[left] at (-0.9,\@karnaughmap@var@mapsizey@*0.5) {\small{#6}};
119     % when there are 2 sub maps
120     \ifnum\@karnaughmap@var@mapsizez@=2
121       \node[above] at (7,4.9) {\small{#5}};
122       % extra sub maps labels
123       \node[below] at (2,-0.1) {\small{#7$=0$}};
124       \node[below] at (7,-0.1) {\small{#7$=1$}};
125     \fi
126     % when there are 4 sub maps
127     \ifnum\@karnaughmap@var@mapsizez@=4
128       \node[above] at (7,4.9) {\small{#5}};
129       \node[left] at (-0.9,-3) {\small{#6}};
130       % extra sub maps labels
131       \node[below] at (2,-0.1) {\small{#7$=00$}};
132       \node[below] at (7,-0.1) {\small{#7$=01$}};
133       \node[below] at (2,-5.1) {\small{#7$=10$}};
134       \node[below] at (7,-5.1) {\small{#7$=11$}};
135     \fi
136     % data
137     \matrix[
138       matrix of nodes,
139       ampersand replacement=\&,
140       column sep={1cm,between origins},
141       row sep={1cm,between origins},
142   ] at (\@karnaughmap@var@mapsizex@*0.5+\@karnaughmap@local@maprealignmentx@,\@karnaughmap@var@map
143       \@karnaughmap@local@matrixtemplate@%
```

```
144       };
145 }{
146     \end{tikzpicture}
147   \endgroup
148 }
```

File 238 **lwarp-keyfloat.sty**

§ 347 Package keyfloat

(*Emulates or patches code by* Brian Dunn.)

keyfloat (*Pkg*) keyfloat is supported with a considerable amount of hacking. (It's a mashup of lwarp, keyfloat, and tocdata.)

⚠ keywrap If placing a `\keyfig[H]` inside a keywrap, use an absolute width for `\keyfig`, instead of `lw`-proportional widths. (The `[H]` option forces the use of a minipage, which internally adjusts for a virtual 6-inch wide minipage, which then corrupts the `lw` option.)

For wrapped figures, overhang and number of lines are ignored.

for HTML output:

```
1 \LWR@ProvidesPackagePass{keyfloat}[2019/09/23]
2
3 \IfPackageAtLeastTF{keyfloat}{2019/09/23}{\relax}{
4     \PackageError{lwarp-keyfloat}
5     {%
6         The keyfloat package is out of date.\MessageBreak
7         Update to keyfloat v2.01 2019/09/23 or later%
8     }
9     {%
10        Please update the keyfloat package.  It's worth it!%
11    }
12 }
```

After keyfloat has loaded:

```
13 \AtBeginDocument{
```

\KFLT@LWR@hook@boxouter (*Hook*) [keyfloat] Integration for keyfloat.

```
14 \providecommand*{\KFLT@LWR@hook@boxouter}{}
15
16 \renewcommand*{\KFLT@LWR@hook@boxouter}{%
17     \ifbool{KFLT@keywrap}{%
18     }{%
19         \ifnumequal{\value{KFLT@keyfloatdepth}}{0}{%
20             \setlength{\linewidth}{6in}%
21             \setlength{\textwidth}{6in}%
22             \setlength{\textheight}{9in}%
23         }{}%
24     }%
25     \normalcolor%
26 }
```

\KFLT@LWR@hook@keysubfloats (*Hook*) [keyfloat] Integration for keyfloat.

```
\LetLtxMacro\KFLT@LWR@hook@keysubfloats\KFLT@LWR@hook@boxouter
```

\KFLT@LWR@hook@keyfloatsminipage (*Hook*) [keyfloat]

Integration for keyfloat.

```
\let\KFLT@LWR@hook@keyfloatsminipage\relax
\let\endKFLT@LWR@hook@keyfloatsminipage\relax
\newenvironment*{KFLT@LWR@hook@keyfloatsminipage}[1]{}{}
```

\KFLT@LWR@hook@keyfloats (*Hook*) [keyfloat]

Integration for keyfloat.

```
\LetLtxMacro\KFLT@LWR@hook@keyfloats\KFLT@LWR@hook@boxouter

\renewcommand*{\KFLT@maybeendfloatrow}{%
    \ifnumless{\value{KFLT@thiscol}}{\value{KFLT@numcols}}%
        {}% thiscol < numcols
        {% >=
            \defcounter{KFLT@thiscol}{0}%
        }%
}%

\renewcommand{\KFLT@trackrows}%
{%
```

If are nested inside a keyfloats or a subfloat:

```
    \ifboolexpr{%
        test {\ifnumgreater{\value{KFLT@keyfloatdepth}}{0}} or%
        bool{KFLT@inkeysubfloats}%
    }%
    {% nested
```

Tracks row start and end:

```
        \KFLT@maybestartfloatrow%
```

Possibly fill space between columns:

```
        \ifnumgreater{\value{KFLT@thiscol}}{1}%
            {%
%                 \hfill%
            }%
            {}%
    }% nested
    {}% not nested
}

\RenewDocumentCommand{\KFLT@onefigureimage}{m}
{%
\LWR@traceinfo{KFLT@onefigureimage}%
% \begin{lrbox}{\KFLT@envbox}%
\ifthenelse{\NOT\equal{\KFLT@lw}{}}%
    {%

        \ifdimgreater{\KFLT@h}{0pt}%
        {%
            \KFLT@frame{%
                \includegraphics%
```

```
                [%
                    scale=\KFLT@s,%
                    width=\KFLT@imagewidth,%
                    height=\KFLT@h,%
                    \KFLT@keepaspectratio,%
                ]{#1}%
            }%
        }%
        {%
            \KFLT@frame{\includegraphics%
            [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
        }%
    }%
    {% not linewidth
        \ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
        {% width is given
            \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
            {% w and h
                \KFLT@frame{\includegraphics[%
                    scale=\KFLT@s,%
                    width=\KFLT@imagewidth,%
                    height=\KFLT@h,%
                    \KFLT@keepaspectratio,%
                ]{#1}}%
            }% w and h
            {% only w
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
            }% only w
        }% width is given
        {% width is not given
            \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
            {%
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s,height=\KFLT@h]{#1}}%
            }%
            {%
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s]{#1}}%
            }%
        }% width is not given
    }% not linewidth
% \end{lrbox}%
% \unskip%
% \KFLT@findenvboxwidth%
% \begin{turn}{\KFLT@r}%
% \KFLT@frame{\usebox{\KFLT@envbox}}%
% \unskip%
% \end{turn}%
\LWR@traceinfo{KFLT@onefigureimage: done}%
}

\RenewDocumentEnvironment{KFLT@boxinner}{}
{%
    \LWR@traceinfo{KFLT@boxinner}%
    \LWR@stoppars%
    \minipagefullwidth%
    \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
        \fminipage{\KFLT@imagewidth}%
    }{%
```

```
        \minipage{\KFLT@imagewidth}%
    }%
}
{%
    \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
        \endfminipage%
    }{%
        \endminipage%
    }%
    \LWR@startpars%
    \LWR@traceinfo{KFLT@boxinner: done}%
}

\newcommand*{\LWR@KFLT@settextalign}[1]{%
    \def\LWR@KFLT@textalign{justify}%
    \ifcsstring{KFLT@#1textalign}{\centering}%
        {\def\LWR@KFLT@textalign{center}}%
        {}%
    \ifcsstring{KFLT@#1textalign}{\raggedleft}%
        {\def\LWR@KFLT@textalign{right}}%
        {}%
    \ifcsstring{KFLT@#1textalign}{\raggedright}%
        {\def\LWR@KFLT@textalign{left}}%
        {}%
}

\renewcommand{\KFLT@addtext}[1]
{%
```

Is there text to add?

```
    \ifcsempty{KFLT@#1t}%
    {}% no text
    {% text to add
        {% local
```

Add some space, then create a to contain the text:

```
        \addvspace{\smallskipamount}%
        \LWR@KFLT@settextalign{#1}%
        \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Set the alignment and some text parameters:

```
%         \csuse{KFLT@#1textalign}%
%         \footnotesize%
        \setlength{\parskip}{1.5ex}%
        \setlength{\parindent}{0em}%
```

Typeset the actual text:

```
        \csuse{KFLT@#1t}%
```

Close it all out with a little more space:

```
        \end{BlockClass}%
%         \par\addvspace{2ex}%
        }% local
```

```
    }% text to add
}

\IfPackageLoadedTF{tocdata}
{}
{% tocdata not loaded

    \newcommand*{\LWR@KFLT@setnamealign}[1]{%
        \def\LWR@KFLT@textalign{justify}%
        \ifstrequal{#1}{\centering}%
            {\def\LWR@KFLT@textalign{center}}%
            {}%
        \ifstrequal{#1}{\raggedleft}%
            {\def\LWR@KFLT@textalign{right}}%
            {}%
        \ifstrequal{#1}{\raggedright}%
            {\def\LWR@KFLT@textalign{left}}%
            {}%
    }

    \renewcommand*{\KFLT@@addartisttext}[3]{%
```

Add space and create the name inside a :

```
%         \addvspace{\medskipamount}%
%     \begin{minipage}{\linewidth}%
        \LWR@KFLT@setnamealign{#3}%
        \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Text alignment is #3, and depends on artist or author:

```
%     #3%
```

#1 is empty or 'subgrp'
#2 is empty for artist, 'u' for author:

```
        \footnotesize\textsc{%
            \KFLT@optionalname{\csuse{KFLT@#1a#2p}}%
            \KFLT@optionalname{\csuse{KFLT@#1a#2f}}%
            \csuse{KFLT@#1a#2l}%
            \csuse{KFLT@#1a#2s}%
        }%
%     \end{minipage}%
        \end{BlockClass}
%         \par\addvspace{2ex}%
    }

}% tocdata not loaded
```

Env KFLT@marginfloat [⟨*offset*⟩] {⟨*type*⟩}

```
\DeclareDocumentEnvironment{KFLT@marginfloat}{O{-1.2ex} m}
{%
    \uselengthunit{PT}%
    \LWR@BlockClassWP%
        {float:right; width:2in; margin:10pt}%
        {}%
        (note)%
        {marginblock}%
```

```
    \renewcommand*{\@captype}{#2}%
    \minipage{1.2\LWR@usersmarginparwidth}%
    \setlength{\marginparwidth}{.95\LWR@usersmarginparwidth}%
}
{%
    \endminipage%
    \endLWR@BlockClassWP%
}

\DeclareDocumentEnvironment{marginfigure}{o}
    {\begin{KFLT@marginfloat}{figure}}
    {\end{KFLT@marginfloat}}

\DeclareDocumentEnvironment{margintable}{o}
    {\begin{KFLT@marginfloat}{table}}
    {\end{KFLT@marginfloat}}
```

Env keywrap {⟨*width*⟩} {⟨*keyfloat*⟩}

```
\DeclareDocumentEnvironment{keywrap}{m +m}
{%
    \begin{LWR@setvirtualpage}*
    \setlength{\LWR@templengthone}{#1}%
    \begin{LWR@BlockClassWP}%
        {%
        float:right; width:\LWR@printlength{\LWR@templengthone};  % extra space
            margin:10pt%
        }%
        {}%
        (note)%
        {marginblock}%
    \setlength{\linewidth}{.95\LWR@templengthone}%
    \booltrue{KFLT@keywrap}%
    #2%
    \end{LWR@BlockClassWP}%
    \end{LWR@setvirtualpage}%
}
{}

}% AtBeginDocument
```

File 239 **lwarp-keystroke.sty**

§ 348 Package

keystroke

(*Emulates or patches code by* Werner Fink.)

keystroke (*Pkg*) keystroke is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{keystroke}[2010/04/23]

\newcommand*{\LWR@HTML@keystroke}[1]{
    \InlineClass{keystroke}{#1}
}
\LWR@formatted{keystroke}

```

```
\newcommand*{\LWR@HTML@Return}{\keystroke{\HTMLunicode{021A9}}}
\LWR@formatted{Return}

\newcommand*{\LWR@HTML@BSpace}{\keystroke{\HTMLunicode{027FB}}}
\LWR@formatted{BSpace}

\newcommand*{\LWR@HTML@Tab}{\keystroke{|\HTMLunicode{021C6}|}}
\LWR@formatted{Tab}

\newcommand*{\LWR@HTML@UArrow}{\keystroke{\HTMLunicode{02191}}}
\LWR@formatted{UArrow}

\newcommand*{\LWR@HTML@DArrow}{\keystroke{\HTMLunicode{02193}}}
\LWR@formatted{DArrow}

\newcommand*{\LWR@HTML@LArrow}{\keystroke{\HTMLunicode{02190}}}
\LWR@formatted{LArrow}

\newcommand*{\LWR@HTML@RArrow}{\keystroke{\HTMLunicode{02192}}}
\LWR@formatted{RArrow}

% Preserves the language options:
\LetLtxMacro\LWR@HTML@Shift\Shift
\xpatchcmd{\LWR@HTML@Shift}
    {$\Uparrow$}
    {\HTMLunicode{21D1}}
    {}
    {}
\LWR@formatted{Shift}

\LetLtxMacro\LWR@HTML@PgUp\PgUp
\xpatchcmd{\LWR@HTML@PgUp}
    {$\uparrow$}
    {\HTMLunicode{2191}}
    {}
    {}
\LWR@formatted{PgUp}

\LetLtxMacro\LWR@HTML@PgDown\PgDown
\xpatchcmd{\LWR@HTML@PgDown}
    {$\downarrow$}
    {\HTMLunicode{2193}}
    {}
    {}
\LWR@formatted{PgDown}
```

File 240 **lwarp-kpfonts.sty**

§ 349 Package kpfonts

(*Emulates or patches code by* Christophe Caignaert.)

kpfonts (*Pkg*) kpfonts is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation honors the options uprightRoman for \D only, classicReIm, frenchstyle for Greek only, upright for Greek only, uprightgreeks, slantedGreeks, and mathcalasscript.

The dedicated macros for Greek work correctly.

svg math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{kpfonts}[2010/08/20]

\LWR@infoprocessingmathjax{kpfonts}

\LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

\ifkp@calasscr
    \CustomizeMathJax{\let\LWRorigmathscr\mathscr}
    \CustomizeMathJax{\let\LWRorigmathcal\mathcal}
    \CustomizeMathJax{\let\mathscr\LWRorigmathcal}
    \CustomizeMathJax{\let\mathcal\LWRorigmathscr}
\fi

\ifkp@upgrk % lowercase
    \LWR@mathjax@addgreek@l@up{}{}
    \LWR@mathjax@addgreek@l@it{other}{}
\else
    \LWR@mathjax@addgreek@l@up{other}{}
\fi

\ifkp@slGrk
    \LWR@mathjax@addgreek@u@it*{}{}
    \LWR@mathjax@addgreek@u@up*{other}{}
    \LWR@mathjax@addgreek@u@up*{var}{}
\else
    \LWR@mathjax@addgreek@u@it*{other}{}
    \LWR@mathjax@addgreek@u@it*{var}{}
\fi

\LWR@mathjax@addgreek@u@up*{}{up}
\LWR@mathjax@addgreek@l@up{}{up}

\LWR@mathjax@addgreek@u@it*{}{sl}
\LWR@mathjax@addgreek@l@it{}{sl}

\CustomizeMathJax{\newcommand{\partialsl}{\mathord{\unicode{x1D715}}}}
\CustomizeMathJax{\let\partialup\uppartial}% not upright

\ifkp@oldReIm
\else
    \CustomizeMathJax{\renewcommand{\Re}{\mathfrak{Re}}}
    \CustomizeMathJax{\renewcommand{\Im}{\mathfrak{Im}}}
\fi

\ifkp@Dcommand
    \ifkp@upRm%
        \CustomizeMathJax{
            \def\D#1{\mathclose{\,\mathrm{d}}#1}
        }
    \else
        \CustomizeMathJax{
            \def\D#1{\mathclose{\,\mathit{d}}#1}
```

```
        }
    \fi
\fi

\CustomizeMathJax{\let\pounds\mathsterling}
\CustomizeMathJax{\let\kppounds\mathsterling}

\CustomizeMathJax{\newcommand{\mathup}[1]{\mathrm{#1}}}% never sans
\CustomizeMathJax{\let\mathupright\mathup}

\end{warpMathJax}
```

File 241 **lwarp-kpfonts-otf.sty**

§ 350 Package

kpfonts-otf

(*Emulates or patches code by* Daniel Flipo.)

kpfonts-otf (*Pkg*) kpfonts-otf is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation honors the options `fancyReIm`, `mathcal`, `frenchstyle` for Greek only, and `mathcalasscript`.

Also see the options for unicode-math, which is loaded by kpfonts-otf.

The unicode-math dedicated macros for Greek work correctly.

⚠ `\mathversion` The MathJax emulation does not change with the use of `\mathversion`. Whatever emulation is established at the begin of the document will remain.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{kpfonts-otf}[2020/06/20]

\LWR@infoprocessingmathjax{kpfonts-otf}

\LWR@origRequirePackage{lwarp-common-mathjax-nonunicode}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

\ifkp@calasscr
    \CustomizeMathJax{\let\mathscr\mathcal}
\else
    \CustomizeMathJax{\let\mathcal\mathscr}
\fi

\ifkp@frenchstyle
    \LWR@mathjax@addgreek@l@up{}{}
    \LWR@mathjax@addgreek@u@up*{}{}
\fi

\ifkp@oldReIm
    \CustomizeMathJax{\renewcommand{\Re}{\mathfrak{Re}}}
    \CustomizeMathJax{\renewcommand{\Im}{\mathfrak{Im}}}
\else
\fi
```

```
27
28 \ifkp@Dcommand
29     \CustomizeMathJax{
30         \def\D#1{\mathclose{\,\mathrm{d}}#1}
31     }
32 \fi
33
34 \CustomizeMathJax{\let\varint\int}
35 \CustomizeMathJax{\let\variint\iint}
36 \CustomizeMathJax{\let\variiint\iiint}
37 \CustomizeMathJax{\let\variiiint\iiiint}
38 \CustomizeMathJax{\let\varidotsint\idotsint}
39
40 \CustomizeMathJax{\newcommand{\varointctrclockwise}{%
41     \mathop{\unicode{x2939}\!\!\unicode{x0222E}}%
42 }}
43
44 \CustomizeMathJax{\newcommand{\oiintclockwise}{%
45     \mathop{\unicode{x0222F}\!\!\unicode{x2938}}%
46 }}
47
48 \CustomizeMathJax{\newcommand{\oiintctrclockwise}{%
49     \mathop{\unicode{x2939}\!\!\unicode{x0222F}}%
50 }}
51
52 \CustomizeMathJax{\newcommand{\varoiintclockwise}{%
53     \mathop{\unicode{x0222F}\!\!\unicode{x2938}}%
54 }}
55
56 \CustomizeMathJax{\newcommand{\varoiintctrclockwise}{%
57     \mathop{\unicode{x2939}\!\!\unicode{x0222F}}%
58 }}
59
60 \CustomizeMathJax{\newcommand{\oiiintclockwise}{%
61     \mathop{\unicode{x02230}\!\!\unicode{x2938}}%
62 }}
63
64 \CustomizeMathJax{\newcommand{\oiiintctrclockwise}{%
65     \mathop{\unicode{x2939}\!\!\unicode{x02230}}%
66 }}
67
68 \CustomizeMathJax{\newcommand{\varoiiintclockwise}{%
69     \mathop{\unicode{x02230}\!\!\unicode{x2938}}%
70 }}
71
72 \CustomizeMathJax{\newcommand{\varoiiintctrclockwise}{%
73     \mathop{\unicode{x2939}\!\!\unicode{x02230}}%
74 }}
75
76 \CustomizeMathJax{\newcommand{\sqiint}{%
77     \mathop{\unicode{x2A16}\!\!\unicode{x2A16}}%
78 }}
79
80 \CustomizeMathJax{\newcommand{\sqiiint}{%
81     \mathop{\unicode{x2A16}\!\!\unicode{x2A16}\!\!\unicode{x2A16}}%
82 }}
83
84 \CustomizeMathJax{\let\widearc\overparen}
85 \CustomizeMathJax{\let\widearcarrow\overrightarrow}
86 \CustomizeMathJax{\let\overrightarc\overrightarrow}
```

```
87
88 \end{warpMathJax}
```

File 242 **lwarp-layaureo.sty**

§ 351 Package layaureo

layaureo (*Pkg*) layaureo is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{layaureo}[2004/09/16]
```

File 243 **lwarp-layout.sty**

§ 352 Package layout

layout (*Pkg*) layout is ignored.

for HTML output: Discard all options for lwarp-layout:

```
1 \LWR@ProvidesPackageDrop{layout}[2014/10/28]

2 \NewDocumentCommand{\layout}{s}{}
```

File 244 **lwarp-layouts.sty**

§ 353 Package layouts

layouts (*Pkg*) layouts is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{layouts}[2009/09/02]

2 \newif\ifoddpagelayout
3   \oddpagelayouttrue
4 \newif\iftwocolumnlayout
5   \twocolumnlayoutfalse
6 \newif\ifdrawmarginpars
7   \drawmarginparstrue
8 \newif\ifdrawparameters
9   \drawparameterstrue
10 \newif\iflistaspara
11   \listasparatrue
12 \newif\ifruninhead
13   \runinheadfalse
14 \newif\ifprintparameters
15   \printparameterstrue
16 \newif\ifdrawdimensions
17   \drawdimensionsfalse
18 \newif\ifprintheadings
19   \printheadingstrue
20 \newcommand{\testdrawdimensions}{}
21 \newcommand{\testprintparameters}{}
```

```
\newcommand{\setlabelfont}[1]{}
\newcommand{\setparametertextfont}[1]{}
\newcommand{\setvaluestextsize}[1]{}
\newcommand{\setlayoutscale}[1]{}
\newcommand{\setuplayouts}{}
\newcommand{\printinunitsof}[1]{}
\newcommand{\prntlen}[1]{}
\newcommand{\trypaperwidth}[1]{}
\newcommand{\trypaperheight}[1]{}
\newcommand{\tryhoffset}[1]{}
\newcommand{\tryvoffset}[1]{}
\newcommand{\trytopmargin}[1]{}
\newcommand{\tryheadheight}[1]{}
\newcommand{\tryheadsep}[1]{}
\newcommand{\trytextheight}[1]{}
\newcommand{\tryfootskip}[1]{}
\newcommand{\tryoddsidemargin}[1]{}
\newcommand{\tryevensidemargin}[1]{}
\newcommand{\trytextwidth}[1]{}
\newcommand{\trymarginparsep}[1]{}
\newcommand{\trymarginparwidth}[1]{}
\newcommand{\trymarginparpush}[1]{}
\newcommand{\trycolumnsep}[1]{}
\newcommand{\trycolumnseprule}[1]{}
\newcommand{\setfootbox}[2]{}
\newcommand{\currentpage}{}
\newcommand{\drawpage}{(draw page)}
\newcommand{\pagediagram}{(page diagram)}
\newcommand{\pagedesign}{(page design)}
\newcommand{\pagevalues}{(page values)}
\newcommand{\trystockwidth}[1]{}
\newcommand{\trystockheight}[1]{}
\newcommand{\trytrimedge}[1]{}
\newcommand{\trytrimtop}[1]{}
\newcommand{\tryuppermargin}[1]{}
\newcommand{\tryspinemargin}[1]{}
\newcommand{\currentstock}{}
\newcommand{\drawstock}{(draw stock)}
\newcommand{\stockdiagram}{(stock diagram)}
\newcommand{\stockdesign}{(stock design)}
\newcommand{\stockvalues}{(stock values)}
\newcommand{\tryitemindent}[1]{}
\newcommand{\trylabelwidth}[1]{}
\newcommand{\trylabelsep}[1]{}
\newcommand{\tryleftmargin}[1]{}
\newcommand{\tryrightmargin}[1]{}
\newcommand{\trylistparindent}[1]{}
\newcommand{\trytopsep}[1]{}
\newcommand{\tryparskip}[1]{}
\newcommand{\trypartopsep}[1]{}
\newcommand{\tryparsep}[1]{}
\newcommand{\tryitemsep}[1]{}
\newcommand{\currentlist}{}
\newcommand{\drawlist}{(draw list)}
\newcommand{\listdiagram}{(list diagram)}
\newcommand{\listdesign}{(list design)}
\newcommand{\listvalues}{(list values)}
\newcommand{\tryfootins}[1]{}
\newcommand{\tryfootnotesep}[1]{}
\newcommand{\tryfootnotebaseline}[1]{}
```

```
\newcommand{\tryfootruleheight}[1]{}
\newcommand{\tryfootrulefrac}[1]{}
\newcommand{\currentfootnote}{}
\newcommand{\drawfootnote}{(draw footnote)}
\newcommand{\footnotediagram}{(footnote diagram)}
\newcommand{\footnotedesign}{(footnote design)}
\newcommand{\footnotevalues}{(footnote values)}
\newcommand{\tryparindent}[1]{}
\newcommand{\tryparlinewidth}[1]{}
\newcommand{\tryparbaselineskip}[1]{}
\newcommand{\currentparagraph}{}
\newcommand{\drawparagraph}{(draw paragraph)}
\newcommand{\paragraphdiagram}{(paragraph diagram)}
\newcommand{\paragraphdesign}{(paragraph design)}
\newcommand{\paragraphvalues}{(paragraph values)}
\newcommand{\trybeforeskip}[1]{}
\newcommand{\tryafterskip}[1]{}
\newcommand{\tryindent}[1]{}
\newcommand{\currentheading}{}
\newcommand{\drawheading}[1]{(draw heading)}
\newcommand{\headingdiagram}[1]{(heading diagram)}
\newcommand{\headingdesign}[1]{(heading design)}
\newcommand{\headingvalues}{(heading values)}
\newcommand{\trytextfloatsep}[1]{}
\newcommand{\tryfloatsep}[1]{}
\newcommand{\tryintextsep}[1]{}
\newcommand{\trytopfigrule}[1]{}
\newcommand{\trybotfigrule}[1]{}
\newcommand{\currentfloat}{}
\newcommand{\drawfloat}{(draw float)}
\newcommand{\floatdiagram}{(float diagram)}
\newcommand{\floatdesign}{(float design)}
\newcommand{\floatvalues}{(float values)}
\newcommand{\trytotalnumber}[1]{}
\newcommand{\trytopnumber}[1]{}
\newcommand{\trybottomnumber}[1]{}
\newcommand{\trytopfraction}[1]{}
\newcommand{\trytextfraction}[1]{}
\newcommand{\trybottomfraction}[1]{}
\newcommand{\currentfloatpage}{}
\newcommand{\drawfloatpage}{(draw floatpage)}
\newcommand{\floatpagediagram}{(floatpage diagram)}
\newcommand{\floatpagedesign}{(floatpage design)}
\newcommand{\floatpagevalues}{(floatpage values)}
\newcommand{\trytocindent}[1]{}
\newcommand{\trytocnumwidth}[1]{}
\newcommand{\trytoclinewidth}[1]{}
\newcommand{\trytocrmarg}[1]{}
\newcommand{\trytocpnumwidth}[1]{}
\newcommand{\trytocdotsep}[1]{}
\newcommand{\currenttoc}{}
\newcommand{\drawtoc}{(draw toc)}
\newcommand{\tocdiagram}{(toc diagram)}
\newcommand{\tocdesign}{(toc design)}
\newcommand{\tocvalues}{(toc values)}
\newcommand{\drawaspread}[8][0]{(a spread)}
\newcommand{\drawfontframe}[1]{(font frame)}
\newcommand{\drawfontframelabel}[1]{}
```

File 245 **lwarp-leading.sty**

§ 354 Package leading

`leading` (*Pkg*) leading is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{leading}[2008/12/11]
\newcommand\leading[1]{}
```

File 246 **lwarp-leftidx.sty**

§ 355 Package leftidx

(*Emulates or patches code by* Harald Harders.)

`leftidx` (*Pkg*) leftidx works as-is with svg math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{leftidx}[2003/09/24]
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\leftidx}[3]{{\vphantom{#2}}#1#2#3}}
\CustomizeMathJax{\newcommand{\ltrans}[1]{\leftidx{^\mathrm{t}}{\!#1}{}}}
\end{warpMathJax}
```

File 247 **lwarp-letterspace.sty**

§ 356 Package letterspace

(*Emulates or patches code by* R Schlicht.)

`letterspace` (*Pkg*) letterspace is a subset of microtype, which is pre-loaded by lwarp. All user options and macros are ignored and disabled.

for HTML output: Discard all options for lwarp-letterspace:

```
\LWR@ProvidesPackageDrop{letterspace}[2018/01/14]
\newcommand*\lsstyle{}
\newcommand\textls[2][]{}
\def\textls#1#{}
\newcommand*\lslig[1]{#1}
```

File 248 **lwarp-lettrine.sty**

§ 357 Package lettrine

(*Emulates or patches code by* Daniel Flipo.)

`lettrine` (*Pkg*) lettrine is emulated.

for HTML output: Discard all options for lwarp-lettrine:

```
\LWR@ProvidesPackageDrop{lettrine}[2018-08-28]
```

The initial letter is in a of class `lettrine`, and the following text is in a of class `lettrinetext`. `\lettrine` [⟨*keys*⟩] {⟨*letter*⟩} {⟨*additional text*⟩}

```
\DeclareDocumentCommand{\lettrine}{o m m}{%
    \InlineClass{lettrine}{#2}\InlineClass{lettrinetext}{#3} % extra space
}

\newcounter{DefaultLines}
\setcounter{DefaultLines}{2}
\newcounter{DefaultDepth}
\newcommand*{\DefaultOptionsFile}{\relax}
\newcommand*{\DefaultLoversize}{0}
\newcommand*{\DefaultLraise}{0}
\newcommand*{\DefaultLhang}{0}
\newdimen\DefaultFindent
\setlength{\DefaultFindent}{\z@}
\newdimen\DefaultNindent
\setlength{\DefaultNindent}{0.5em}
\newdimen\DefaultSlope
\setlength{\DefaultSlope}{\z@}
\newdimen\DiscardVskip
\setlength{\DiscardVskip}{0.2\p@}
\newif\ifLettrineImage
\newif\ifLettrineOnGrid
\newif\ifLettrineRealHeight

\newcommand*{\LettrineTextFont}{\scshape}
\newcommand*{\LettrineFontHook}{}
\newcommand*{\LettrineFont}[1]{\InlineClass{lettrine}{#1}}
\newcommand*{\LettrineFontEPS}[1]{\includegraphics[height=1.5ex]{#1}}
```

File 249 **lwarp-libertinust1math.sty**

§ 358 Package

libertinust1math

(*Emulates or patches code by* Michael Sharpe.)

`libertinust1math` (*Pkg*) libertinust1math is used as-is for SVG math, and is emulated for MathJax.

The MathJax emulation honors `frenchmath` for Greek but not Latin characters, and `slantedGreek`, `uprightGreek`, and `ISO` also adjust Greek characters. MathJax cannot yet honor options for adjusting Latin characters.

The dedicated macros for upright and italic Greek letters do work correctly.

Some of the symbol font macros such as `\mathsfbf` do not use a sans font because MathJax does not yet have sans Greek.

SVG math honors all font choices, and should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{libertinust1math}[2020/06/10]

\LWR@infoprocessingmathjax{libertinust1math}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

\iflibus@slantedG
    \LWR@mathjax@addgreek@u@it*{}{}
\else
    \LWR@mathjax@addgreek@u@up*{}{}
\fi

\LWR@mathjax@addgreek@u@it*{}{it}
\LWR@mathjax@addgreek@u@up*{up}{}
\LWR@mathjax@addgreek@u@up*{}{up}

\iflibus@frenchm
    \LWR@mathjax@addgreek@l@up{}{}
\else
    \LWR@mathjax@addgreek@l@it{}{}
\fi

\LWR@mathjax@addgreek@l@it{}{it}
\LWR@mathjax@addgreek@l@up{}{up}
\LWR@mathjax@addgreek@l@up{up}{}

\CustomizeMathJax{\let\uppartial\partial}% not upright


\CustomizeMathJax{\let\mathsfbf\mathbf}% not sans
% \CustomizeMathJax{\newcommand{\mathsfbf}[1]{%
%     \mmlToken{mi}[mathvariant="bold-sans-serif"]{#1}% not greek
% }}% not sans

% \CustomizeMathJax{\newcommand{\mathbfit}[1]{\boldsymbol{#1}}}
\CustomizeMathJax{\let\mathbfit\boldsymbol}


% \CustomizeMathJax{\newcommand{\mathsfbfit}[1]{\boldsymbol{#1}}}% not sans
\CustomizeMathJax{\let\mathsfbfit\mathbfit}% not sans
% \CustomizeMathJax{\newcommand{\mathsfbfit}[1]{%
%     \mmlToken{mi}[mathvariant="sans-serif-bold-italic"]{#1}% not greek
% }}%


\CustomizeMathJax{\let\mathsfit\mathit}% not sans
% \CustomizeMathJax{\newcommand{\mathsfit}[1]{%
%     \mmlToken{mi}[mathvariant="sans-serif-italic"]{#1}% not greek
% }}

\CustomizeMathJax{\let\vectorsym\mathbfit}
\CustomizeMathJax{\let\matrixsym\mathbfit}
\CustomizeMathJax{\let\tensorsym\mathsfbfit}
\CustomizeMathJax{\let\mathboldsans\mathsfbfit}
\CustomizeMathJax{\let\mathbold\mathbfit}
```

`lwarp_mathjax.txt` adds `\left`/`\right` support for delimiters.

```
\CustomizeMathJax{\let\dlb\lBrack}
\CustomizeMathJax{\let\drb\rBrack}

\CustomizeMathJax{\let\sqrtsign\sqrt}

\CustomizeMathJax{\let\smallintsl\smallint}
\CustomizeMathJax{\newcommand{\smalliintsl}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\smalliiintsl}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\smalliiiintsl}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\newcommand{\smallointsl}{\mathop{\unicode{x222E}}\limits}}
\CustomizeMathJax{\newcommand{\smalloiintsl}{\mathop{\unicode{x222F}}\limits}}

\CustomizeMathJax{\let\smallintup\smallint}
\CustomizeMathJax{\newcommand{\smalliintup}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\smalliiintup}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\smalliiiintup}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\newcommand{\smallointup}{\mathop{\unicode{x222E}}\limits}}
\CustomizeMathJax{\newcommand{\smalloiintup}{\mathop{\unicode{x222F}}\limits}}

\CustomizeMathJax{\let\intslop\int}
\CustomizeMathJax{\newcommand{\iintslop}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiintslop}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\iiiintslop}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\let\ointslop\oint}
\CustomizeMathJax{\newcommand{\oiintslop}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiintslop}{\mathop{\unicode{x2230}}\limits}}

\CustomizeMathJax{\let\intupop\int}
\CustomizeMathJax{\newcommand{\iintupop}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiintupop}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\iiiintupop}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\let\ointupop\oint}
\CustomizeMathJax{\newcommand{\oiintupop}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiintupop}{\mathop{\unicode{x2230}}\limits}}

\CustomizeMathJax{\newcommand{\smalliint}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\smalliiint}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\smalliiiint}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\newcommand{\smalloint}{\mathop{\unicode{x222E}}\limits}}
\CustomizeMathJax{\newcommand{\smalloiint}{\mathop{\unicode{x222F}}\limits}}

\CustomizeMathJax{\let\intop\int}
\CustomizeMathJax{\newcommand{\iintop}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiintop}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\iiiintop}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\let\ointop\oint}
\CustomizeMathJax{\newcommand{\oiintop}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiintop}{\mathop{\unicode{x2230}}\limits}}

\CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}

\CustomizeMathJax{\newcommand{\bigcupdot}{\mathop{\unicode{x2A03}}}}
\CustomizeMathJax{\newcommand{\bigsqcap}{\mathop{\unicode{x2A05}}}}
\CustomizeMathJax{\newcommand{\xsol}{\mathop{\unicode{x29F8}}}}
\CustomizeMathJax{\newcommand{\xbsol}{\mathop{\unicode{x29F9}}}}
\CustomizeMathJax{\let\prodop\prod}
\CustomizeMathJax{\let\coprodop\coprod}
\CustomizeMathJax{\let\sumop\sum}
\CustomizeMathJax{\let\bigwedgeop\bigwedge}
\CustomizeMathJax{\let\bigveeop\bigvee}
```

```
\CustomizeMathJax{\let\bigcapop\bigcap}
\CustomizeMathJax{\let\bigcupop\bigcup}
\CustomizeMathJax{\let\xsolop\xsol}
\CustomizeMathJax{\let\xbsolop\xbsol}
\CustomizeMathJax{\let\bigodotop\bigodot}
\CustomizeMathJax{\let\bigoplusop\bigoplus}
\CustomizeMathJax{\let\bigotimesop\bigotimes}
\CustomizeMathJax{\let\bigcupdotop\bigcupdot}
\CustomizeMathJax{\let\biguplusop\biguplus}
\CustomizeMathJax{\let\bigsqcapop\bigsqcap}
\CustomizeMathJax{\let\bigsqcupop\bigsqcup}

\CustomizeMathJax{\newcommand{\ovhook}[1]{\mathord{#1\unicode{x00309}}}}
\CustomizeMathJax{\newcommand{\candra}[1]{\mathord{#1\unicode{x00310}}}}
\CustomizeMathJax{\newcommand{\oturnedcomma}[1]{\mathord{#1\unicode{x00312}}}}
\CustomizeMathJax{\newcommand{\ocommatopright}[1]{\mathord{#1\unicode{x00315}}}}
\CustomizeMathJax{\newcommand{\droang}[1]{\mathord{#1\unicode{x0031A}}}}
\CustomizeMathJax{\newcommand{\leftharpoonaccent}[1]{\mathord{#1\unicode{x020D0}}}}
\CustomizeMathJax{\newcommand{\rightharpoonaccent}[1]{\mathord{#1\unicode{x020D1}}}}
\CustomizeMathJax{\newcommand{\leftarrowaccent}[1]{\mathord{#1\unicode{x020D0}}}}
\CustomizeMathJax{\let\rightarrowaccent\vec}

\CustomizeMathJax{\newcommand{\leftrightarrowaccent}[1]{\mathord{#1\unicode{x020E1}}}}
\CustomizeMathJax{\newcommand{\annuity}[1]{\mathord{#1\unicode{x020E7}}}}
\CustomizeMathJax{\newcommand{\widebridgeabove}[1]{\mathord{#1\unicode{x020E9}}}}
\CustomizeMathJax{\newcommand{\asteraccent}[1]{\mathord{#1\unicode{x020F0}}}}

% neutralized:
\CustomizeMathJax{\newcommand{\braceld}{}}
\CustomizeMathJax{\newcommand{\bracerd}{}}
\CustomizeMathJax{\newcommand{\bracelu}{}}
\CustomizeMathJax{\newcommand{\braceru}{}}
\CustomizeMathJax{\newcommand{\braceex}{}}
\CustomizeMathJax{\newcommand{\bracemu}{}}
\CustomizeMathJax{\newcommand{\bracemd}{}}
\CustomizeMathJax{\newcommand{\parenld}{}}
\CustomizeMathJax{\newcommand{\parenrd}{}}
\CustomizeMathJax{\newcommand{\parenlu}{}}
\CustomizeMathJax{\newcommand{\parenru}{}}
\CustomizeMathJax{\newcommand{\bracketld}{}}
\CustomizeMathJax{\newcommand{\bracketrd}{}}
\CustomizeMathJax{\newcommand{\bracketlu}{}}
\CustomizeMathJax{\newcommand{\bracketru}{}}
\CustomizeMathJax{\newcommand{\bracketex}{}}
\CustomizeMathJax{\newcommand{\parenex}{}}

\CustomizeMathJax{\newcommand{lhook}{~}}
\CustomizeMathJax{\newcommand{rhook}{~}}
\CustomizeMathJax{\newcommand{relbar}{-}}
\CustomizeMathJax{\newcommand{Relbar}{=}}

\CustomizeMathJax{\newcommand{\mapstochar}{\mathrel{\unicode{x21A6}}}}

\CustomizeMathJax{\newcommand{\Zbar}{\mathord{\unicode{x0001B5}}}}
\CustomizeMathJax{\newcommand{\notchar}{\mathrel{\unicode{x000AC}}}}
\CustomizeMathJax{\newcommand{\upbackepsilon}{\mathord{\unicode{x03F6}}}}
\CustomizeMathJax{\newcommand{\smblkcircle}{\mathbin{\unicode{x02022}}}}
\CustomizeMathJax{\newcommand{\enleadertwodots}{\mathord{\unicode{x02025}}}}
\CustomizeMathJax{\newcommand{\unicodeellipsis}{\mathord{\unicode{x02026}}}}
\CustomizeMathJax{\newcommand{\mathellipsis}{\mathinner{\unicode{x02026}}}}
```

```
\CustomizeMathJax{\newcommand{\dprime}{\mathord{\unicode{x02033}}}}
\CustomizeMathJax{\newcommand{\trprime}{\mathord{\unicode{x02034}}}}
\CustomizeMathJax{\newcommand{\backdprime}{\mathord{\unicode{x02036}}}}
\CustomizeMathJax{\newcommand{\backtrprime}{\mathord{\unicode{x02037}}}}
\CustomizeMathJax{\newcommand{\caretinsert}{\mathord{\unicode{x02038}}}}
\CustomizeMathJax{\newcommand{\Exclam}{\mathord{\unicode{x0203C}}}}

\CustomizeMathJax{\newcommand{\hyphenbullet}{\mathord{\unicode{x02043}}}}
\CustomizeMathJax{\newcommand{\fracslash}{\mathbin{\unicode{x02044}}}}
\CustomizeMathJax{\newcommand{\Question}{\mathord{\unicode{x02047}}}}
\CustomizeMathJax{\newcommand{\closure}{\mathrel{\unicode{x02050}}}}
\CustomizeMathJax{\newcommand{\qprime}{\mathord{\unicode{x02057}}}}
\CustomizeMathJax{\newcommand{\vertoverlay}{\mathrel{\unicode{x020D2}}}}
\CustomizeMathJax{\newcommand{\enclosecircle}{\mathord{\unicode{x020DD}}}}
\CustomizeMathJax{\newcommand{\enclosesquare}{\mathord{\unicode{x020DE}}}}
\CustomizeMathJax{\newcommand{\enclosetriangle}{\mathord{\unicode{x020E4}}}}
\CustomizeMathJax{\newcommand{\Eulerconst}{\mathord{\unicode{x02107}}}}
\CustomizeMathJax{\newcommand{\turnediota}{\mathord{\unicode{x02129}}}}
\CustomizeMathJax{\newcommand{\Angstrom}{\mathord{\unicode{x0212B}}}}

\CustomizeMathJax{\newcommand{\sansLturned}{\mathord{\unicode{x02142}}}}
\CustomizeMathJax{\newcommand{\sansLmirrored}{\mathord{\unicode{x02143}}}}
\CustomizeMathJax{\newcommand{\Yup}{\mathord{\unicode{x02144}}}}
\CustomizeMathJax{\newcommand{\upand}{\mathbin{\unicode{x0214B}}}}
\CustomizeMathJax{\newcommand{\increment}{\mathord{\unicode{x02206}}}}
\CustomizeMathJax{\newcommand{\smallin}{\mathrel{\unicode{x0220A}}}}
\CustomizeMathJax{\newcommand{\nni}{\mathrel{\unicode{x0220C}}}}

\CustomizeMathJax{\newcommand{\smallni}{\mathrel{\unicode{x0220D}}}}
\CustomizeMathJax{\newcommand{\QED}{\mathord{\unicode{x0220E}}}}
\CustomizeMathJax{\newcommand{\vysmwhtcircle}{\mathbin{\unicode{x02218}}}}
\CustomizeMathJax{\newcommand{\vysmblkcircle}{\mathbin{\unicode{x02219}}}}
\CustomizeMathJax{\newcommand{\rightangle}{\mathord{\unicode{x0221F}}}}

\CustomizeMathJax{\newcommand{\Colon}{\mathrel{\unicode{x02237}}}}
\CustomizeMathJax{\newcommand{\dotminus}{\mathbin{\unicode{x02238}}}}
\CustomizeMathJax{\newcommand{\dashcolon}{\mathrel{\unicode{x02239}}}}
\CustomizeMathJax{\newcommand{\dotsminusdots}{\mathrel{\unicode{x0223A}}}}
\CustomizeMathJax{\newcommand{\kernelcontraction}{\mathrel{\unicode{x0223B}}}}
\CustomizeMathJax{\newcommand{\invlazys}{\mathbin{\unicode{x0223E}}}}

\CustomizeMathJax{\newcommand{\sinewave}{\mathord{\unicode{x0223F}}}}
\CustomizeMathJax{\newcommand{\nsime}{\mathrel{\unicode{x02244}}}}
\CustomizeMathJax{\newcommand{\simneqq}{\mathrel{\unicode{x02246}}}}
\CustomizeMathJax{\newcommand{\napprox}{\mathrel{\unicode{x02249}}}}
\CustomizeMathJax{\newcommand{\approxident}{\mathrel{\unicode{x0224B}}}}
\CustomizeMathJax{\newcommand{\backcong}{\mathrel{\unicode{x0224C}}}}

\CustomizeMathJax{\newcommand{\nasymp}{\mathrel{\unicode{x0226D}}}}
\CustomizeMathJax{\newcommand{\nlesssim}{\mathrel{\unicode{x02274}}}}
\CustomizeMathJax{\newcommand{\ngtrsim}{\mathrel{\unicode{x02275}}}}
\CustomizeMathJax{\newcommand{\nlessgtr}{\mathrel{\unicode{x02278}}}}
\CustomizeMathJax{\newcommand{\ngtrless}{\mathrel{\unicode{x02279}}}}

\CustomizeMathJax{\newcommand{\nsubset}{\mathrel{\unicode{x02284}}}}
\CustomizeMathJax{\newcommand{\nsupset}{\mathrel{\unicode{x02285}}}}

\CustomizeMathJax{\newcommand{\cupleftarrow}{\mathbin{\unicode{x0228C}}}}
\CustomizeMathJax{\newcommand{\cupdot}{\mathbin{\unicode{x0228D}}}}
\CustomizeMathJax{\newcommand{\circledequal}{\mathbin{\unicode{x0229C}}}}
```

```
231
232 \CustomizeMathJax{\newcommand{\assert}{\mathrel{\unicode{x022A6}}}}
233 \CustomizeMathJax{\newcommand{\VDash}{\mathrel{\unicode{x022AB}}}}
234 \CustomizeMathJax{\newcommand{\prurel}{\mathrel{\unicode{x022B0}}}}
235
236 \CustomizeMathJax{\newcommand{\origof}{\mathrel{\unicode{x022B6}}}}
237 \CustomizeMathJax{\newcommand{\smallprod}{\mathop{\unicode{x0220F}}}}% not small
238 \CustomizeMathJax{\newcommand{\smallcoprod}{\mathop{\unicode{x02210}}}}% not small
239 \CustomizeMathJax{\newcommand{\smallsum}{\mathop{\unicode{x02211}}}}% not small
240 \CustomizeMathJax{\newcommand{\Hfraktur}{\mathord{\unicode{x1D525}}}}
241 \CustomizeMathJax{\newcommand{\dsol}{\mathbin{\unicode{x029F6}}}}
242 \CustomizeMathJax{\newcommand{\rsolbar}{\mathbin{\unicode{x029F7}}}}
243
244 \CustomizeMathJax{\newcommand{\eqless}{\mathrel{\unicode{x022DC}}}}
245 \CustomizeMathJax{\newcommand{\eqgtr}{\mathrel{\unicode{x022DD}}}}
246 \CustomizeMathJax{\newcommand{\npreccurlyeq}{\mathrel{\unicode{x022E0}}}}
247 \CustomizeMathJax{\newcommand{\nsucccurlyeq}{\mathrel{\unicode{x022E1}}}}
248 \CustomizeMathJax{\newcommand{\nsqsubseteq}{\mathrel{\unicode{x022E2}}}}
249 \CustomizeMathJax{\newcommand{\nsqsupseteq}{\mathrel{\unicode{x022E3}}}}
250 \CustomizeMathJax{\newcommand{\sqsubsetneq}{\mathrel{\unicode{x022E4}}}}
251 \CustomizeMathJax{\newcommand{\sqsupsetneq}{\mathrel{\unicode{x022E5}}}}
252 \CustomizeMathJax{\newcommand{\nvartriangleleft}{\mathrel{\unicode{x022EA}}}}
253 \CustomizeMathJax{\newcommand{\nvartriangleright}{\mathrel{\unicode{x022EB}}}}
254
255 \CustomizeMathJax{\newcommand{\vdotsmath}{\mathrel{\unicode{x022EE}}}}
256 \CustomizeMathJax{\newcommand{\unicodecdots}{\mathord{\unicode{x022EF}}}}
257 \CustomizeMathJax{\newcommand{\adots}{\mathrel{\unicode{x022F0}}}}
258 \CustomizeMathJax{\newcommand{\succneq}{\mathrel{\unicode{x02AB2}}}}
259 \CustomizeMathJax{\newcommand{\preceqq}{\mathrel{\unicode{x02AB3}}}}
260 \CustomizeMathJax{\newcommand{\succeqq}{\mathrel{\unicode{x02AB4}}}}
261 \CustomizeMathJax{\newcommand{\precneq}{\mathrel{\unicode{x02AB1}}}}
262
263 \CustomizeMathJax{\newcommand{\mapsfrom}{\mathrel{\unicode{x021A4}}}}
264
265 \CustomizeMathJax{\newcommand{\longmapsfrom}{\mathrel{\unicode{x027FB}}}}
266
267 \CustomizeMathJax{\newcommand{\diameter}{\mathord{\unicode{x02300}}}}
268 \CustomizeMathJax{\newcommand{\coloneq}{\mathrel{\unicode{x02254}}}}
269 \CustomizeMathJax{\newcommand{\eqcolon}{\mathrel{\unicode{x02255}}}}
270 \CustomizeMathJax{\newcommand{\arceq}{\mathrel{\unicode{x02258}}}}
271 \CustomizeMathJax{\newcommand{\wedgeq}{\mathrel{\unicode{x02259}}}}
272 \CustomizeMathJax{\newcommand{\veeeq}{\mathrel{\unicode{x0225A}}}}
273
274 \CustomizeMathJax{\newcommand{\stareq}{\mathrel{\unicode{x0225B}}}}
275 \CustomizeMathJax{\newcommand{\eqdef}{\mathrel{\unicode{x0225D}}}}
276 \CustomizeMathJax{\newcommand{\measeq}{\mathrel{\unicode{x0225E}}}}
277 \CustomizeMathJax{\newcommand{\questeq}{\mathrel{\unicode{x0225F}}}}
278 \CustomizeMathJax{\newcommand{\nequiv}{\mathrel{\unicode{x02262}}}}
279 \CustomizeMathJax{\newcommand{\Equiv}{\mathrel{\unicode{x02263}}}}
280
281 \CustomizeMathJax{\newcommand{\house}{\mathord{\unicode{x02302}}}}
282
283 \CustomizeMathJax{\newcommand{\musicalnote}{\mathord{\unicode{x0266A}}}}
284 \CustomizeMathJax{\newcommand{\degree}{\mathord{\unicode{x000B0}}}}
285 \CustomizeMathJax{\newcommand{\mathsection}{\mathord{\unicode{x000A7}}}}
286 \CustomizeMathJax{\newcommand{\mathparagraph}{\mathord{\unicode{x000B6}}}}
287 \CustomizeMathJax{\newcommand{\checkmarkmath}{\mathord{\unicode{x02713}}}}
288 \CustomizeMathJax{\newcommand{\invnot}{\mathord{\unicode{x02310}}}}
289
290 \CustomizeMathJax{\newcommand{\mathvisiblespace}{\mathord{\unicode{x02423}}}}
```

```
\CustomizeMathJax{\newcommand{\mdlgblksquare}{\mathord{\unicode{x025A0}}}}
\CustomizeMathJax{\newcommand{\mdlgwhtsquare}{\mathord{\unicode{x025A1}}}}

\CustomizeMathJax{\newcommand{\bigblacktriangleup}{\mathord{\unicode{x025B2}}}}
\CustomizeMathJax{\newcommand{\varbigtriangleup}{\mathord{\unicode{x025B3}}}}

\CustomizeMathJax{\newcommand{\bigblacktriangledown}{\mathord{\unicode{x025BC}}}}
\CustomizeMathJax{\newcommand{\varbigtriangledown}{\mathord{\unicode{x025BD}}}}
\CustomizeMathJax{\newcommand{\Longmapsfrom}{\mathrel{\unicode{x027FD}}}}

% bug in print font:
\CustomizeMathJax{\newcommand{\mdlgblkdiamond}{\mathord{\unicode{x025C6}}}}

\CustomizeMathJax{\newcommand{\mdlgwhtdiamond}{\mathord{\unicode{x025C7}}}}
\CustomizeMathJax{\newcommand{\Longmapsto}{\mathrel{\unicode{x027FE}}}}
\CustomizeMathJax{\newcommand{\fisheye}{\mathord{\unicode{x025C9}}}}
\CustomizeMathJax{\newcommand{\mdlgwhtlozenge}{\mathord{\unicode{x025CA}}}}
\CustomizeMathJax{\newcommand{\mdlgwhtcircle}{\mathbin{\unicode{x025CB}}}}
\CustomizeMathJax{\newcommand{\bullseye}{\mathord{\unicode{x025CE}}}}
\CustomizeMathJax{\newcommand{\mdlgblkcircle}{\mathord{\unicode{x025CF}}}}

\CustomizeMathJax{\newcommand{\Nwarrow}{\mathrel{\unicode{x021D6}}}}
\CustomizeMathJax{\newcommand{\Nearrow}{\mathrel{\unicode{x021D7}}}}
\CustomizeMathJax{\newcommand{\Searrow}{\mathrel{\unicode{x021D8}}}}
\CustomizeMathJax{\newcommand{\Swarrow}{\mathrel{\unicode{x021D9}}}}

\CustomizeMathJax{\newcommand{\Mapsfrom}{\mathord{\unicode{x02906}}}}
\CustomizeMathJax{\newcommand{\smwhtcircle}{\mathord{\unicode{x025E6}}}}
\CustomizeMathJax{\newcommand{\smwhtdiamond}{\mathbin{\unicode{x022C4}}}}
\CustomizeMathJax{\newcommand{\Mapsto}{\mathord{\unicode{x02907}}}}

\CustomizeMathJax{\let\ngets\nleftarrow}
\CustomizeMathJax{\let\nsimeq\nsime}
\CustomizeMathJax{\let\nle\nleq}
\CustomizeMathJax{\let\nge\ngeq}

\end{warpMathJax}
```

File 250 **lwarp-lineno.sty**

§ 359 Package

lineno

(*Emulates or patches code by* Stephan I. Böttcher.)

`lineno` (*Pkg*) lineno is partly emulated, but mostly ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lineno}[2005/11/02]

\newcommand*\resetlinenumber[1][\@ne]{}

\def\linenumbers{%
    \@ifnextchar[{\resetlinenumber}%]
                 {\@ifstar{\resetlinenumber}{}}%
    }

\newcommand*{\nolinenumbers}{}

```

```
\@namedef{linenumbers*}{\par\linenumbers*}
\@namedef{runninglinenumbers*}{\par\runninglinenumbers*}

\def\endlinenumbers{\par}
\let\endrunninglinenumbers\endlinenumbers
\let\endpagewiselinenumbers\endlinenumbers
\expandafter\let\csname endlinenumbers*\endcsname\endlinenumbers
\expandafter\let\csname endrunninglinenumbers*\endcsname\endlinenumbers
\let\endnolinenumbers\endlinenumbers

\def\pagewiselinenumbers{\linenumbers\setpagewiselinenumbers}

\def\runninglinenumbers{\setrunninglinenumbers\linenumbers}

\def\setpagewiselinenumbers{}

\def\setrunninglinenumbers{}

\def\linenomath{}%
\@namedef{linenomath*}{}%
\def\endlinenomath{}
\expandafter\let\csname endlinenomath*\endcsname\endlinenomath

\let\linelabel\label

\def\switchlinenumbers{\@ifstar{}{}}
\def\setmakelinenumbers#1{\@ifstar{}{}}

\def\leftlinenumbers{\@ifstar{}{}}
\def\rightlinenumbers{\@ifstar{}{}}

\newcounter{linenumber}
\newcount\c@pagewiselinenumber
\let\c@runninglinenumber\c@linenumber

\def\runningpagewiselinenumbers{}
\def\realpagewiselinenumbers{}


\NewDocumentCommand\modulolinenumbers{s o}{}

\chardef\c@linenumbermodulo=5
\modulolinenumbers[1]

\newcommand*\firstlinenumber[1]{}

\newcommand\internallinenumbers{}
\let\endinternallinenumbers\endlinenumbers
\@namedef{internallinenumbers*}{\internallinenumbers}
\expandafter\let\csname endinternallinenumbers*\endcsname\endlinenumbers

\newcommand*{\linenoplaceholder}[1]{% redefine per language
    (line number reference for \detokenize\expandafter{#1})
}

\newcommand*{\lineref}[2][]{\linenoplaceholder{#2}}
\newcommand*{\linerefp}[2][]{\linenoplaceholder{#2}}
\newcommand*{\linerefr}[2][]{\linenoplaceholder{#2}}

\newcommand\quotelinenumbers
```

```
   {\@ifstar\linenumbers{\@ifnextchar[\linenumbers{\linenumbers*}}}

\newdimen\linenumbersep
\newdimen\linenumberwidth
\newdimen\quotelinenumbersep

\quotelinenumbersep=\linenumbersep
\let\quotelinenumberfont\linenumberfont

\def\linenumberfont{\normalfont\tiny\sffamily}


\linenumberwidth=10pt
\linenumbersep=10pt

\def\thelinenumber{}

\def\LineNumber{}
\def\makeLineNumber{}
\def\makeLineNumberLeft{}
\def\makeLineNumberRight{}
\def\makeLineNumberOdd{}
\def\makeLineNumberEven{}
\def\makeLineNumberRunning{}


\newenvironment{numquote}     {\quote}{\endquote}
\newenvironment{numquotation} {\quotation}{\endquotation}
\newenvironment{numquote*}    {\quote}{\endquote}
\newenvironment{numquotation*}{\quotation}{\endquotation}

\newdimen\bframerule
\bframerule=\fboxrule

\newdimen\bframesep
\bframesep=\fboxsep

\newenvironment{bframe}
{%
    \LWR@forceminwidth{\bframerule}%
    \BlockClass[
        border:\LWR@printlength{\LWR@atleastonept} solid black ; %
        padding:\LWR@printlength{\bframesep}%
    ]{bframe}
}
{\endBlockClass}
```

File 251 **lwarp-lips.sty**

§ 360 Package lips

(*Emulates or patches code by* Matt Swift.)

lips (*Pkg*) lips is emulated.

```
% \LWR@ProvidesPackageDrop{lips}
\PackageInfo{lwarp}{Using the lwarp version of package ‘lips’.}%
```

```
\ProvidesPackage{lwarp-lips}[2001/08/31]

\NewDocumentCommand{\Lips}{}{\textellipsis}

\NewDocumentCommand{\BracketedLips}{}{[\textellipsis]}

\let\lips\Lips
\let\olips\lips

\DeclareOption*{}
\DeclareOption{mla}{
\let\lips\BracketedLips
}
\ProcessOptions\relax

\newcommand \LPNobreakList {}
```

File 252 **lwarp-lipsum.sty**

§ 361 Package lipsum

(*Emulates or patches code by* Patrick Happel.)

lipsum (*Pkg*) lipsum is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{lipsum}[2021-03-03]

\SetLipsumParListItemEnd{%
    \LWR@closeparagraph%

    \leavevmode\LWR@orignewline%
}
```

File 253 **lwarp-listings.sty**

§ 362 Package listings

(*Emulates or patches code by* Carsten Heinz, Brooks Moses, Jobst Hoffmann.)

listings (*Pkg*) listings is supported with some limitations. Text formatting and escape characters are not yet supported.

```
\LWR@ProvidesPackagePass{listings}[2018/09/02]
```

Force flexible columns. Fixed columns inserts spaces in the PDF output.

```
\lst@column@flexible
```

Patches to embed listings inside pre tags:

```
\let\LWR@origlst@Init\lst@Init
```

```
\let\LWR@origlst@DeInit\lst@DeInit

\let\LWR@origlsthkEveryPar\lsthk@EveryPar

\renewcommand{\l@lstlisting}[2]{\hypertocfloat{1}{lstlisting}{lol}{#1}{#2}}
```

\lstset {⟨*options*⟩}

Use the listings `literate` option to replace HTML entities:

```
\def\lstset@#1{\endgroup%
%  \ifx\@empty#1%
%         \@empty%
%     \else%
        \setkeys{lst}{%
            #1%
            ,literate=%
            {<}{\HTMLentity{lt}}{4}%
            {>}{\HTMLentity{gt}}{4}%
            {'}{\HTMLentity{apos}}{6}%
            {`}{\HTMLentity{grave}}{7}%
```

The ampersand is not treated here, as the result is inconsistent spacing. It is nevertheless converted to `&` elsewhere. Sanitizing the double quote interferes with listings' conversion of visible spaces inside strings.

```
        }%
%     \fi%
}
```

\lst@Init {⟨*backslash-processing*⟩} Done at the start of a listing.

```
\renewcommand{\lst@Init}[1]{%
```

Perform the listings initialization:

```
\LWR@traceinfo{lst@Init}%
```

`\LWR@forcenewpage` is moved to the start to avoid a spurrious bug with paragraph handling and conditionals.

```
\lst@ifdisplaystyle%        lwarp
    \LWR@forcenewpage%      lwarp
\fi%                        lwarp
```

Escapes do not work yet, and are disabled:

```
\let\lst@ifmathescape\iffalse%      lwarp
\let\lst@DefEsc\relax%              lwarp
\def\lst@escapebegin{}%             lwarp
\def\lst@escapeend{}%               lwarp

\renewcommand*{\@captype}{lstlisting}%      lwarp
    \let\lst@aboveskip\z@\let\lst@belowskip\z@%     lwarp
    \gdef\lst@boxpos{t}%            lwarp
    \let\lst@frame\@empty%          lwarp
    \let\lst@frametshape\@empty%    lwarp
    \let\lst@framershape\@empty%    lwarp
    \let\lst@framebshape\@empty%    lwarp
    \let\lst@framelshape\@empty%    lwarp
    \lstframe@\lst@frameround ffff\relax%   lwarp
    \lst@multicols\@empty%  lwarp
    \begingroup%
```

Inside the listing, temporarily prevent underfull \hbox warnings.

```
            \hbadness=10000\relax%
    \ifx\lst@float\relax\else%
        \edef\@tempa{\noexpand\lst@beginfloat{lstlisting}[\lst@float]}%
        \expandafter\@tempa%
    \fi%
    \ifx\lst@multicols\@empty\else%
        \edef\lst@next{\noexpand\multicols{\lst@multicols}}%
        \expandafter\lst@next%
    \fi%
    \ifhmode\ifinner \lst@boxtrue \fi\fi%
    \lst@ifbox%
        \lsthk@BoxUnsafe%
        \hbox to\z@\bgroup%
             $\if t\lst@boxpos \vtop%
        \else \if b\lst@boxpos \vbox%
        \else \vcenter \fi\fi%
        \bgroup \par\noindent%
    \else%
        \lst@ifdisplaystyle%
            \lst@EveryDisplay%
            \par\penalty-50\relax%
            \vspace\lst@aboveskip%
        \fi%
    \fi%
    \normalbaselines%
    \abovecaptionskip\lst@abovecaption\relax%
    \belowcaptionskip\lst@belowcaption\relax%
    \lst@MakeCaption t%
```

Use the overall listing label instead of the line number label:

```
\LWR@traceinfo{lst@Init: defining current label !\@currentlabel!}%
    \let\LWR@listings@currentlabel\@currentlabel%            lwarp
\LWR@traceinfo{lst@Init: defining current label !\cref@currentlabel!}%
    \let\LWR@listings@cref@currentlabel\cref@currentlabel%   lwarp

\LWR@traceinfo{lst@Init: preinit and init}%
    \lsthk@PreInit \lsthk@Init%

    \let\@currentlabel\LWR@listings@currentlabel%            lwarp
    \let\cref@currentlabel\LWR@listings@cref@currentlabel%   lwarp

\LWR@traceinfo{lst@Init: M}%
    \lst@ifdisplaystyle
        \global\let\lst@ltxlabel\@empty
        \if@inlabel
            \lst@ifresetmargins
                \leavevmode
            \else
                \xdef\lst@ltxlabel{\the\everypar}%
                \lst@AddTo\lst@ltxlabel{%
                    \global\let\lst@ltxlabel\@empty
                    \everypar{\lsthk@EveryLine\lsthk@EveryPar}}%
            \fi
        \fi
        \everypar\expandafter{\lst@ltxlabel
                              \lsthk@EveryLine\lsthk@EveryPar}%
    \else
        \everypar{}
        \let\lst@NewLine\@empty
```

```
    \fi
\LWR@traceinfo{lst@Init: P}%
    \lsthk@InitVars \lsthk@InitVarsBOL
    \lst@Let{13}\lst@MProcessListing
    \let\lst@Backslash#1%
    \lst@EnterMode{\lst@Pmode}{\lst@SelectCharTable}%
    \lst@InitFinalize%
\LWR@traceinfo{lst@Init: S}%
```

Avoids extra horizontal space:

```
\def\lst@framelr{}%     lwarp
```

```
\LWR@traceinfo{lst@Init: finished origlst@Init}%
\lst@ifdisplaystyle%    lwarp
```

Creating a display.

Disable line numbers, produce the <pre>, then reenable line numbers.

```
    \LWR@traceinfo{lst@Init: About to create verbatim.}%  lwarp
    \let\lsthk@EveryPar\relax%                  lwarp

    \LWR@atbeginverbatim{programlisting}%       lwarp

    \let\lsthk@EveryPar\LWR@origlsthkEveryPar%  lwarp
\else%                                          lwarp
```

Inline, so open a :

```
    \ifbool{LWR@verbtags}{\LWR@htmltag{%        lwarp
       span class=\textquotedbl{}inlineprogramlisting\textquotedbl%    lwarp
    }}{}%                                       lwarp
\fi%                                            lwarp
\LWR@traceinfo{lst@Init: done}%
}
```

\lst@DeInit

Done at the end of a listing.

```
\renewcommand*{\lst@DeInit}{%
\LWR@traceinfo{lst@DeInit}%
\lst@ifdisplaystyle%
```

Creating a display.

Disable line numbers, produce the </pre>, then reenable line numbers:

```
    \let\lsthk@EveryPar\relax%
    \LWR@afterendverbatim%
    \let\lsthk@EveryPar\LWR@origlsthkEveryPar%
\else%
```

Inline, so create the closing :

```
    \ifbool{LWR@verbtags}{\noindent\LWR@htmltag{/span}}{}%
\fi%
```

Final listings deinit:

```
    \lst@XPrintToken \lst@EOLUpdate
    \global\advance\lst@newlines\m@ne
    \lst@ifshowlines
        \lst@DoNewLines
    \else
        \setbox\@tempboxa\vbox{\lst@DoNewLines}%
    \fi
```

```
    \lst@ifdisplaystyle \par\removelastskip \fi
    \lsthk@ExitVars\everypar{}\lsthk@DeInit\normalbaselines\normalcolor
    \lst@MakeCaption b%
    \lst@ifbox
        \egroup $\hss \egroup
        \vrule\@width\lst@maxwidth\@height\z@\@depth\z@
    \else
        \lst@ifdisplaystyle
            \par\penalty-50\vspace\lst@belowskip
        \fi
    \fi
    \ifx\lst@multicols\@empty\else
        \def\lst@next{\global\let\@checkend\@gobble
                      \endmulticols
                      \global\let\@checkend\lst@@checkend}
        \expandafter\lst@next
    \fi
    \ifx\lst@float\relax\else
        \expandafter\lst@endfloat
    \fi
    \endgroup
\LWR@traceinfo{lst@DeInit done}%
}
```

\lst@MakeCaption {⟨*t*/*b*⟩}

This is called BOTH at the top and at the bottom of each listing.

Patched for lwarp.

```
\def\lst@MakeCaption#1{%
\LWR@traceinfo{lst@MakeCaption at #1}%
  \lst@ifdisplaystyle
\LWR@traceinfo{lst@MakeCaption: making a listings display caption}%
    \ifx #1t%
         \ifx\lst@@caption\@empty\expandafter\lst@HRefStepCounter \else
                                  \expandafter\refstepcounter
         \fi {lstlisting}%
% \LWR@traceinfo{About to assign label: !\lst@label!}%
%         \ifx\lst@label\@empty\else
% \label{\lst@label}\fi
% \LWR@traceinfo{Finished assigning the label.}%
        \let\lst@arg\lst@intname \lst@ReplaceIn\lst@arg\lst@filenamerpl
        \global\let\lst@name\lst@arg \global\let\lstname\lst@name
        \lst@ifnolol\else
            \ifx\lst@@caption\@empty
                \ifx\lst@caption\@empty
                    \ifx\lst@intname\@empty
                    \else
                        \def\lst@temp{ }%
                        \ifx\lst@intname\lst@temp \else
```

This code places a contents entry for a non-float. This would have to be modified for lwarp:

```
\LWR@traceinfo{lst@MakeCaption: addcontents lst@name: -\lst@name-}%
%                           \addcontentsline{lol}{lstlisting}{\lst@name}
                    \fi
                \fi
            \fi
        \else
```

This would have to be modified for lwarp:

```
\LWR@traceinfo{lst@MakeCaption: addcontents lst@@caption: -\lst@@caption-}%
                  \addcontentsline{lol}{lstlisting}%
{\protect\numberline{\thelstlisting}%
{\protect\ignorespaces \LWR@isolate{\lst@@caption} \protect\relax}}%
            \fi
         \fi
     \fi
    \ifx\lst@caption\@empty\else
\LWR@traceinfo{lst@MakeCaption: lst@caption not empty-}%
        \lst@IfSubstring #1\lst@captionpos
            {\begingroup
\LWR@traceinfo{lst@MakeCaption: at the selected position}%
```

These space and box commands are not needed for HTML output:

```
%              \let\@@vskip\vskip
%              \def\vskip{\afterassignment\lst@vskip \@tempskipa}%
%              \def\lst@vskip{\nobreak\@@vskip\@tempskipa\nobreak}%
%              \par\@parboxrestore\normalsize\normalfont % \noindent (AS)
%              \ifx #1t\allowbreak \fi
             \ifx\lst@title\@empty
```

New lwarp code to create a caption:

```
                    \LWR@stoppars%      lwarp
                \lst@makecaption\fnum@lstlisting{\ignorespaces \lst@caption}
             \else
```

New lwarp code to create a title:

```
%                  \lst@maketitle\lst@title % (AS)
\LWR@traceinfo{lst@MakeCaption: Making title: \lst@title}%
\begin{BlockClass}{lstlistingtitle}%    lwarp
\lst@maketitle\lst@title%               lwarp
\end{BlockClass}%                       lwarp
             \fi%
\LWR@traceinfo{lst@MakeCaption: About to assign label: !\lst@label!}%
        \ifx\lst@label\@empty\else%
\leavevmode% gets rid of bad space factor error
\GetTitleStringExpand{\lst@caption}%
\edef\LWR@lntemp{\GetTitleStringResult}%
\edef\@currentlabelname{\detokenize\expandafter{\LWR@lntemp}}%
\label{\lst@label}\fi%
\LWR@traceinfo{lst@MakeCaption: Finished assigning the label.}%
```

Not needed for lwarp:

```
%              \ifx #1b\allowbreak \fi
             \endgroup}{}%
    \fi
\LWR@traceinfo{lst@MakeCaption: end of making a listings display caption}%
  \else
\LWR@traceinfo{lst@MakeCaption: INLINE}%
  \fi
\LWR@traceinfo{lst@MakeCaption: done at #1}%
}

\renewcommand{\lst@maketitle}[1]{%
    \LWR@isolate{#1}%
}%

```

line numbers Patched to keep left line numbers outside of the left margin, and place right line numbers in a field \VerbatimHTMLWidth wide.

```
\lst@Key{numbers}{none}{%
    \let\lst@PlaceNumber\@empty
    \lstKV@SwitchCases{#1}%
    {none:\\%
     left:\def\lst@PlaceNumber{%
```

For now, lwarp places left line numbers inline. Ideally the entire line would be moved to the right, but conflicts with list indenting occurs.

```
%         \LWR@origllap{
            \LWR@orignormalfont%
            \lst@numberstyle{\thelstnumber}\kern\lst@numbersep%
%         }
    }\\%
     right:\def\lst@PlaceNumber{\LWR@origrlap{\LWR@orignormalfont
                \kern 6in \kern\lst@numbersep
                \lst@numberstyle{\thelstnumber}}}%
    }{\PackageError{lwarp-listings}{Numbers #1 unknown}\@ehc}}
```

File 254 **lwarp-listliketab.sty**

§ 363 Package

listliketab

listliketab (*Pkg*) listliketab is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{listliketab}[2005/01/09]

\newcommand*{\storestyleof}[1]{}
\newcommand*{\storeliststyle}{}
\newenvironment{listliketab}{}{}
```

File 255 **lwarp-lltjext.sty**

§ 364 Package

lltjext

(*Emulates or patches code by* The LuaTeX-ja project team.)

lltjext (*Pkg*) lltjext is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{lltjext}[2018/10/07]

\protected\def\yoko{%
  \directlua{luatexja.direction.set_list_direction(4, 'yoko')}%
}
\protected\def\tate{\yoko}
\protected\def\dtou{\yoko}
\protected\def\utod{\yoko}

\define@key[ltj]{japaram}{direction}{}

```

```
\yoko

\DeclareExpandableDocumentCommand{\rensuji}{s o m}{#3}

\DeclareDocumentCommand{\layoutfloat}{d() o m}{}

\DeclareDocumentCommand{\DeclareLayoutCaption}{m d<> d() o}{}

\LetLtxMacro\pcaption\caption

\DeclareDocumentCommand{\layoutcaption}{d<> d() o}{}

\let\captiondir\relax
\RenewDocumentEnvironment{LWR@HTML@minipage}{d<> O{t} O{} O{t} m}
    {\LWR@HTML@sub@minipage{#2}{#3}{#4}{#5}}
    {\endLWR@HTML@sub@minipage}

\RenewDocumentCommand{\LWR@HTML@parbox}{d<> O{t} O{} O{t} m +m}
{
\LWR@traceinfo{parbox of width #4}%
\begin{minipage}[#2][#3][#4]{#5}%
#6
\end{minipage}%
}

\RenewDocumentCommand{\pbox}{d<> O{0pt} O{c} m}{%
\global\booltrue{LWR@minipagefullwidth}%
\parbox{#2}{#4}%
}
```

File 256 **lwarp-lltjp-siunitx.sty**

§ 365 Package

lltjp-siunitx

(*Emulates or patches code by* The LuaTeX-ja project team.)

`lltjp-siunitx` (*Pkg*) lltjp-siunitx is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{lltjp-siunitx}% 2021-10-31, no date assigned in file
```

This is the siunitx v3 file, as patched by lltjp-siunitx.

```
\ExplSyntaxOn

\cs_set_protected:Npn \siunitx_print_text:n #1
  {
    \text
      {
        \ltj@allalchar % <--- LuaTeX-ja
        \bool_if:NT \l__siunitx_print_text_family_bool
          { \fontfamily { \familydefault } }
        \bool_if:NT \l__siunitx_print_text_series_bool
          { \fontseries { \seriesdefault } }
        \bool_if:NT \l__siunitx_print_text_shape_bool
          { \fontshape { \shapedefault } }
        \bool_lazy_any:nT
          {
```

```
            { \l__siunitx_print_text_family_bool }
            { \l__siunitx_print_text_series_bool }
            { \l__siunitx_print_text_shape_bool }
          }
          { \selectfont }
        \tl_use:N \l__siunitx_print_text_font_tl
      \exp_args:NnV \tl_if_head_eq_meaning:nNTF {#1} \l_siunitx_unit_fraction_tl
          {
            \__siunitx_print_text_fraction:Nnn #1
          }
          {
            \__siunitx_print_text_replace:n {#1}
          }
      }
  }

\ExplSyntaxOff
```

File 257 **lwarp-lltjp-tascmac.sty**

§ 366 Package

lltjp-tascmac

`lltjp-tascmac` (*Pkg*) lltjp-tascmac is a patch for tascmac, and is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lltjp-tascmac}[2020/12/24]
```

File 258 **lwarp-longtable.sty**

§ 367 Package

longtable

(*Emulates or patches code by* David Carlisle.)

`longtable` (*Pkg*) longtable is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{longtable}[2014/10/28]
```

Use one of either `\endhead` or `\endfirsthead` for both print and HTML, and use a `\warpprintonly` macro to disable the other head phrase, and also the `\endfoot` and `\endfirstfoot` phrases. (See section 8.10.4 if using threeparttablex.)

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  [ <lastfoot macros> ] \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{
  [ <lastfoot macros> ]   % HTML last footer, without \endfoot
                                          % or \endlastfoot.
}
\end{longtable}
```

⚠ Misplaced \noalign — Use the \warpprintonly macro instead of the warpprint environment. Doing so helps avoid "Misplaced \noalign." when using \begin{warpprint}.

⚠ \kill — \kill is ignored, place a \kill line inside

```
\begin{warpprint} . . .  \end{warpprint}
```

or place it inside \warpprintonly.

⚠ lateximage — longtable is not supported inside a lateximage.

See:
http://tex.stackexchange.com/questions/43006/why-is-input-not-expandable

Used to detect more than one of \endhead and \endfirsthead in use for HTML at the same time.

```
\newbool{LWR@longtable@havehead}
\boolfalse{LWR@longtable@havehead}
```

longtable (*env.*) * [⟨*horizalignment*⟩] {⟨*colspec*⟩} Emulates the longtable environment.

Per the caption package, the starred version steps the counter per caption. The unstarred version steps the counter once at the beginning, but not at each caption.

Options [c], [l], and [r] are ignored.

```
\newenvironment{longtable*}[2][]{%
    \LWR@floatbegin{table}%
    \ifdef{\setcaptiontype}{% caption package:
        \setcaptiontype{\LTcaptype}%
        \caption@setoptions{longtable}%
        \caption@setoptions{@longtable}%
        \caption@LT@setup%
    }{% w/o caption package:
        \renewcommand*{\@captype}{\LTcaptype}%
    }%
    \booltrue{LWR@starredlongtable}%
    \boolfalse{LWR@longtable@havehead}%
    \let\captionlistentry\LWR@LTcaptionlistentry%
    \tabular{#2}%
}
{\endtabular\LWR@floatend}

\newenvironment{longtable}[2][]{%
    \LWR@floatbegin{table}%
    \ifdef{\setcaptiontype}{% caption package:
        \setcaptiontype{\LTcaptype}%
        \caption@setoptions{longtable}%
        \caption@setoptions{@longtable}%
        \caption@LT@setup%
    }{% w/o caption package:
        \renewcommand*{\@captype}{\LTcaptype}%
    }%
    \refstepcounter{\LTcaptype}%
    \boolfalse{LWR@longtable@havehead}%
    \let\captionlistentry\LWR@LTcaptionlistentry%
    \tabular{#2}%
```

```
}
{\endtabular\LWR@floatend}
```

Provided for compatibility, but ignored:

```
\newcounter{LTchunksize}
```

Error for heads which should have been in `\warpprintonly`:

```
\newcommand*{\LWR@longtable@headerror}{%
    \PackageError{lwarp-longtable}
    {For longtable:\MessageBreak
    1: Keep either one of an \protect\endhead\space or\MessageBreak
        \space\protect\endfirsthead\space phrase as-is,\MessageBreak
        \space to be used by both print and HTML.\MessageBreak
    2: Place any other \protect\end... phrases inside a\MessageBreak
        \space\protect\warpprintonly\space macro,
            to be ignored by HTML.\MessageBreak
    3: At the end of the table,\MessageBreak
        \space add a final footer for HTML\MessageBreak
        \space inside a \protect\warpHTMLonly\space macro.
            This can be\MessageBreak
        \space a copy of an \protect\endfoot\space or
            \protect\endfirstfoot\MessageBreak
        \space phrase, but without the actual \protect\endfoot\MessageBreak
        \space or \protect\endfirstfoot\space macros.\MessageBreak
        \space If using threeparttablex, add\MessageBreak
        \space \protect\insertTableNotes\space here,
            optionally with\MessageBreak
        \space \protect\UseMinipageWidths\space in front.\MessageBreak
    See the Lwarp documentation regarding\MessageBreak
    longtables and threeparttablex}
    {See the Lwarp documentation regading longtables and threeparttablex.}
}
```

Error if more than one of `\endhead` or `\endfirsthead` is outside of `warpprintonly`.

```
\newcommand*{\LWR@longtable@maybeheaderror}{%
\ifbool{LWR@longtable@havehead}%
    {\LWR@longtable@headerror}%
    {%
        \booltrue{LWR@longtable@havehead}
        \LWR@tabularendofline% throws away options //[dim] and //*
    }%
}
```

Error if more than one of these is outside of `warpprint`.

```
\def\endhead{\LWR@longtable@maybeheaderror}
\def\endfirsthead{\LWR@longtable@maybeheaderror}
```

Error if ANY of these is outside of `warpprint`.

```
\def\endfoot{\LWR@longtable@headerror}
\def\endlastfoot{\LWR@longtable@headerror}
```

```
\let\tabularnewline\\
```

```
\providecommand*{\LWR@HTML@tabularnewline}{\LWR@tabularendofline}
\LWR@formatted{tabularnewline}

\newcommand{\setlongtables}{}% Obsolete command, does nothing.
\newlength{\LTleft}
\newlength{\LTright}
\newlength{\LTpre}
\newlength{\LTpost}
\newlength{\LTcapwidth}

\LetLtxMacro\LWR@origkill\kill
\renewcommand*{\kill}{\LWR@tabularendofline}
\appto\LWR@restoreorigformatting{%
\LetLtxMacro\kill\LWR@origkill%
}
```

File 259 **lwarp-lpic.sty**

§ 368 Package lpic

(*Emulates or patches code by* R. Matveyev.)

lpic (*Pkg*) lpic is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{lpic}[2010/12/23]

\BeforeBeginEnvironment{lpic}{%
    \begin{lateximage}[-lpic-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{lpic}{\end{lateximage}}
```

File 260 **lwarp-lscape.sty**

§ 369 Package lscape

(*Emulates or patches code by* D. P. Carlisle.)

lscape (*Pkg*) lscape is ignored.

for HTML output: Discard all options for lwarp-lscape.

```
\LWR@ProvidesPackageDrop{lscape}[2000/10/22]

\newenvironment*{landscape}{}{}
```

File 261 **lwarp-ltablex.sty**

§ 370 Package ltablex

(*Emulates or patches code by* Anil K. Goel.)

`ltablex` (*Pkg*) ltablex is emulated by lwarp.

for HTML output: Relies on tabularx.

```
\RequirePackage{longtable}
\RequirePackage{tabularx}

\LWR@ProvidesPackageDrop{ltablex}[2014/08/13]

\DeclareDocumentEnvironment{tabularx}{m o m}
{\longtable{#3}}
{\endlongtable}

\DeclareDocumentEnvironment{tabularx*}{m o m}
{\longtable{#3}}
{\endlongtable}

\newcommand*{\keepXColumns}{}
\newcommand*{\convertXColumns}{}
```

File 262 **lwarp-ltcaption.sty**

§ 371 Package ltcaption

(*Emulates or patches code by* Axel Sommerfeldt.)

`ltcaption` (*Pkg*) ltcaption is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{ltcaption}[2018/08/26]
```

`\LTcaptype` is already defined by lwarp.

`longtable*` is already defined by lwarp-longtable.

```
\newlength{\LTcapskip}
\newlength{\LTcapleft}
\newlength{\LTcapright}
\newcommand*{\LTcapmarginsfalse}{}
```

File 263 **lwarp-ltxgrid.sty**

§ 372 Package ltxgrid

`ltxgrid` (*Pkg*) ltxgrid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{ltxgrid}[2010/07/25]

\newcommand*{\onecolumngrid}{}
\newcommand*{\twocolumngrid}{}
\newcommand*{\removestuff}{}
\newcommand*{\addstuff}[2]{}
\newcommand*{\replacestuff}[2]{}
```

File 264 **lwarp-ltxtable.sty**

§ 373 Package

ltxtable

`ltxtable` (*Pkg*) ltxtable is emulated.

⚠ table numbering The print version does not seem to honor `longtable*` from the caption package, while lwarp does.

for HTML output:

```
\RequirePackage{tabularx,longtable}
\LWR@ProvidesPackageDrop{ltxtable}[1995/12/11]
```

`\LTXtable` {⟨*width*⟩} {⟨*file*⟩}

```
\newcommand*{\LTXtable}[2]{%
    \input{#2}%
}
```

File 265 **lwarp-lua-check-hyphen.sty**

§ 374 Package

lua-check-hyphen

`lua-check-hyphen` (*Pkg*) lua-check-hyphen is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lua-check-hyphen}[2018/04/19]
```

```
\newcommand*{\LuaCheckHyphen}[1]{}
```

File 266 **lwarp-lua-visual-debug.sty**

§ 375 Package

lua-visual-debug

`lua-visual-debug` (*Pkg*) lua-visual-debug is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lua-visual-debug}[2016/05/30]
```

File 267 **lwarp-luacolor.sty**

§ 376 Package

luacolor

`luacolor` (*Pkg*) luacolor is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{luacolor}[2016/05/16]
```

```
\newcommand{\luacolorProcessBox}[1]{}
```

File 268 **lwarp-luamplib.sty**

§ 377 Package luamplib

(*Emulates or patches code by* Hans Hagen, Taco Hoekwater, Elie Roux, Philipp Gesang, Kim Dohyun.)

`luamplib` (*Pkg*) luamplib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{luamplib}[2020/02/24]

\BeforeBeginEnvironment{mplibcode}{%
    \begin{lateximage}[-mplibcode-~\PackageDiagramAltText]%
}
\AfterEndEnvironment{mplibcode}{\end{lateximage}}
```

File 269 **lwarp-luatexko.sty**

§ 378 Package luatexko

(*Emulates or patches code by* Dohyun Kim, Soojin Nam.)

`luatexko` (*Pkg*) luatexko is patched for use by lwarp.

Modern HTML is used for `\dotemph`, `\ruby`, and offset and thickness control for `\uline`, etc.

for HTML output:

```
\LWR@ProvidesPackagePass{luatexko}[2021/07/10]

\protected\def\typesetvertical{}
\protected\def\typesethorizontal{}

\def\verticaltypesetting{\BlockClass{verticalrl}}
\def\beginverticaltypesetting{\BlockClass{verticalrl}}
\def\endverticaltypesetting{\endBlockClass}

\protected\def\vertical#1{\BlockClass{verticalrl}}
\protected\def\endvertical{\endBlockClass}
\protected\def\horizontal#1{\BlockClass{horizontaltb}}
\protected\def\endhorizontal{\endBlockClass}
\DeclareDocumentCommand{\vertlatin}{m}{#1}

\newcommand{\LWR@HTML@dotemph}[1]{%
%     \uline{#1}%
    \InlineClass[text-emphasis-style: dot]{dotemph}{#1}%
}
\LWR@formatted{dotemph}

\newcommand{\LWR@HTML@ruby}[2]{%
    \LWR@htmltagc{ruby}%
    #1%
    \LWR@htmltagc{rp}(\LWR@htmltagc{/rp}%
```

```
    \LWR@htmltagc{rt}#2\LWR@htmltagc{/rt}%
    \LWR@htmltagc{rp})\LWR@htmltagc{/rp}%
    \LWR@htmltagc{/ruby}%
}
\LWR@formatted{ruby}
```

The following is modified from lwarp-ulem:

```
\NewDocumentCommand{\LWR@HTML@uline}{+m}{%
    \InlineClass%
        (text-decoration:underline; text-decoration-skip: auto)%
        [%
            text-underline-offset: \ulinedown ;
            text-decoration-thickness: \ulinewidth%
        ]%
        {uline}{\LWR@isolate{#1}}%
}
\LWR@formatted{uline}

\NewDocumentCommand{\LWR@HTML@uuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline; text-decoration-skip: auto;%
            text-decoration-style:double%
        )%
        [%
            text-underline-offset: \ulinedown ;
            text-decoration-thickness: \ulinewidth%
        ]%
        {uuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{uuline}

\NewDocumentCommand{\LWR@HTML@uwave}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline; text-decoration-skip: auto;%
            text-decoration-style:wavy%
        )%
        [%
            text-underline-offset: \ulinedown ;
            text-decoration-thickness: \ulinewidth%
        ]%
        {uwave}{\LWR@isolate{#1}}%
}
\LWR@formatted{uwave}

\NewDocumentCommand{\LWR@HTML@sout}{+m}{%
    \InlineClass%
        (text-decoration:line-through)%
        [text-decoration-thickness: \ulinewidth]%
        {sout}{\LWR@isolate{#1}}%
}
\LWR@formatted{sout}

\NewDocumentCommand{\LWR@HTML@xout}{+m}{%
    \InlineClass%
        (text-decoration:line-through)%
        [text-decoration-thickness: \ulinewidth]%
        {xout}{\LWR@isolate{#1}}%
```

```
}
\LWR@formatted{xout}

\NewDocumentCommand{\LWR@HTML@dashuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline;%
            text-decoration-skip: auto;%
            text-decoration-style:dashed%
        )%
        [%
            text-underline-offset: \ulinedown ;
            text-decoration-thickness: \ulinewidth%
        ]%
        {dashuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{dashuline}

\NewDocumentCommand{\LWR@HTML@dotuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline;%
            text-decoration-skip: auto;%
            text-decoration-style: dotted%
        )%
        [%
            text-underline-offset: \ulinedown ;
            text-decoration-thickness: \ulinewidth%
        ]%
        {dotuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{dotuline}
```

File 270 **lwarp-luatodonotes.sty**

§ 379 Package

luatodonotes

(*Emulates or patches code by* Fabian Lipp.)

luatodonotes (*Pkg*) luatodonotes is emulated.

The documentation for todonotes and luatodonotes have an example with a todo inside a caption. If this example does not work it will be necessary to move the todo outside of the caption.

for HTML output:

```
\LWR@ProvidesPackagePass{luatodonotes}[2017/09/30]
```

Nullify options:

```
\@todonotes@additionalMarginEnabledfalse

\if@todonotes@disabled
\else

\newcommand{\ext@todo}{tdo}

\renewcommand{\l@todo}[2]{\hypertocfloat{1}{todo}{ldo}{#1}{#2}}
```

```
\let\LWRTODONOTES@orig@todototoc\todototoc

\renewcommand*{\todototoc}{%
\LWR@phantomsection%
\LWRTODONOTES@orig@todototoc%
}


\renewcommand{\@todonotes@drawMarginNoteWithLine}{%
\fcolorbox
    {\@todonotes@currentbordercolor}
    {\@todonotes@currentbackgroundcolor}
    {\arabic{@todonotes@numberoftodonotes}}
\marginpar{\@todonotes@drawMarginNote}
}

\renewcommand{\@todonotes@drawInlineNote}{%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \if@todonotes@authorgiven%
        {\@todonotes@author:\,}%
        \fi%
        \@todonotes@text%
    }%
}

\newcommand{\@todonotes@drawMarginNote}{%
    \if@todonotes@authorgiven%
        \@todonotes@author\par%
    \fi%
    \arabic{@todonotes@numberoftodonotes}: %
    \fcolorbox%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \@todonotes@sizecommand%
        \@todonotes@text %
    }%
}%

\renewcommand{\missingfigure}[2][]{%
\setkeys{todonotes}{#1}%
\addcontentsline{tdo}{todo}{\@todonotes@MissingFigureText: #2}%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentfigcolor}%
    {%
        \setlength{\fboxrule}{4pt}%
        \fcolorbox{red}{white}{Missing figure} \quad #2%
    }
}

\LetLtxMacro\LWRTODONOTES@orig@todocommon\@todocommon

\RenewDocumentCommand{\@todocommon}{m m}{%
\begingroup%
\renewcommand*{\phantomsection}{}%
\LWRTODONOTES@orig@todocommon{#1}{#2}%
```

```
\endgroup%
}

\renewcommand{\@todoarea}[3][]{%
    \@todonotes@areaselectedtrue%
    \@todocommon{#1}{#2}%
    \todonotes@textmark@highlight{#3}%
    \zref@label{@todonotes@\arabic{@todonotes@numberoftodonotes}@end}%
}%


\DeclareDocumentCommand{\todonotes@textmark@highlight}{m}{%
\InlineClass[background:\LWR@origpound{}B3FFB3]{highlight}{#1}%
}

\fi% \if@todonotes@disabled
```

File 271 **lwarp-luavlna.sty**

§ 380 Package luavlna

(*Emulates or patches code by* Michal Hoftich, Miro Hrončok.)

luavlna (*Pkg*) luavlna is patched for use by lwarp.

The package is disabled for HTML output, due to incompatibilities with lwarp's handling of math SVG images.

for HTML output:

```
\LWR@ProvidesPackagePass{luavlna}[2019/10/30]

\preventsingleoff
\LetLtxMacro\preventsingleon\preventsingleoff
```

File 272 **lwarp-lyluatex.sty**

§ 381 Package lyluatex

(*Emulates or patches code by* Fr. Jacques Peron, Urs Liska, Br. Samuel Springuel.)

lyluatex (*Pkg*) lyluatex is patched for use by lwarp.

For the first compile, to set *lwarpmk*'s configuration, use:

```
lualatex --shell-escape <filename>
```

⚠ images After compiling the document with **lwarpmk html**, use **lwarpmk limages** to convert the Lilypond images for HTML.

CSS The option insert=systems results in an image per system. Each music image "system" is placed inside a of class lyluatex, which defaults to display: inline-block.

⚠ insert=fullpage

CSS The option insert=fullpage results in a single image per page of printed output. Each music "fullpage" image is placed inside a of class lyluatex. To

match the number of measures per line with the printed version, use the geometry package to select the page geometry, or use the lyluatex options for page and staff sizes.

⚠ options To use `\linewidth` or `\textwidth` inside the package options for lyluatex, use the kvoptions-patch package first:

```
\usepackage{kvoptions-patch}
\usepackage[...,line-width-0.8\linewidth,...]{lyluatex}
```

⚠ raw-pdf If using `raw-pdf`, the resulting PDF images must be converted to SVG:

Enter ⇒ **`lwarpmk pdftosvg tmp-ly/*.pdf`**

for HTML output:

```
\LWR@origRequirePackage{luacode}

\LWR@ProvidesPackagePass{lyluatex}[2022/11/07]
```

User-redefinable `ALT` tag:

```
\newcommand*{\LyluatexImageAltText}{-lilypond-~\PackageDiagramAltText}
```

`\ly@compilescore` {⟨*Lilypond object*⟩}

```
\LetLtxMacro\LWR@orig@ly@compilescore\ly@compilescore

\renewcommand*{\ly@compilescore}[1]{%
```

A local group holds a number of changes:

```
    \begingroup%
```

The user's original geometry and font size are restored to match the print version. This allows for correct spacing in the musical score.

```
    \LWR@maybe@orignewpage%
    \LWR@origloadgeometry{LWR@usergeometry}%
    \LWR@print@normalsize%
```

A local group holds a redefined `\includegraphics` which is used by *lyluatex.lua* to insert the *Lilypond* score if `insert=systems` is used. This is now placed inside a `lateximage`, which itself is placed inside a of class `lyluatex`.

`\LWR@addbaselinemarker` preserves the left margins.

```
    \renewcommand{\includegraphics}[2][]{%
        \InlineClass{lyluatex}{%
            \begin{lateximage}[\LyluatexImageAltText]%
            \LWR@addbaselinemarker%
            \LWR@origincludegraphics{##2}%
            \end{lateximage}%
        }%
    }%
```

From the original:

```
    \ly@setunits%
    \setluaoption{ly}{currfiledir}{\currfiledir}
    \setluaoption{ly}{twoside}{\ly@istwosided}
    \directlua{
        #1
        ly.newpage_if_fullpage()
```

```
    }%
    \ly@resetunits%
    \ly@currentfonts%
```

The fullpage version is set inside an HTML :

```
    \directlua{
        if (ly.score.insert == 'fullpage') then
            tex.print{[[\string\begin{BlockClass}{lyluatex}]]}
        end
    }%
```

Generate the score:

```
    \directlua{ly.score:process()}%
```

Close the :

```
    \directlua{
        if (ly.score.insert == 'fullpage') then
            tex.print{[[\string\end{BlockClass}]]}
        end
    }%
```

Move to a new page and renew the regular page geometry:

```
    \LWR@maybe@orignewpage%
    \LWR@origrestoregeometry%
```

End of the local group.

```
    \endgroup%
}
```

In HTML the following generates an error, so is removed:

```
\xpatchcmd{\endly@bufferenv}
    {\hspace{0pt}\\}
    {}
    {}
    {\LWR@patcherror{lyluatex}{endly@bufferenv}}
```

File 273 **lwarp-magaz.sty**

§ 382 Package

magaz

magaz (*Pkg*) magaz is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{magaz}[2011/11/24]

\newcommand\FirstLine[1]{%
    \begingroup%
    \FirstLineFont{%
        \LWR@textcurrentcolor{%
            \LWR@textcurrentfont{%
                #1%
            }%
        }%
    }%
    \endgroup%
}
```

```
13
14 \providecommand\FirstLineFont{\scshape}
```

File 274 **lwarp-makeidx.sty**

§ 383 Package

makeidx

(*Emulates or patches code by* LaTeX Project Team.)

makeidx (*Pkg*) makeidx is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{makeidx}[2014/09/29]
```

\@wrindex is redefined \AtBeginDocument by the lwarp core.

\printindex

```
2 \preto\printindex{%
3     \LWR@maybe@orignewpage%
4     \LWR@startpars%
5 }
```

File 275 **lwarp-manyfoot.sty**

§ 384 Package

manyfoot

manyfoot (*Pkg*) manyfoot is emulated.

bigfoot, manyfoot ⚠ verbatim

Verbatim footnotes are not yet supported.

If using the bigfoot package, and possibly also manyfoot, problems may occur with counter allocation because lwarp uses many counters, and there is a difference in how counters numbered 256 and up are handled in pdf LaTeX. With bigfoot this has been known to show up as an error related to one footnote insert being forbidden inside another. Another problem showed up as a input stack error, and which of these problems occurred depended on how many counters were allocated.

As a possible solution, try creating several new counters before defining bigfoot or manyfoot footnotes, hoping to shift the problematic counter above the 256 threshold. It may instead be necessary to use XeLaTeX or LuaLaTeX instead of pdf LaTeX.

lwarp's emulation of bigfoot uses manyfoot, so some of the bigfoot enhancements are included here.

The bigfoot "default" footnote is ignored, using the lwarp version instead.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{manyfoot}[2005/09/11]

2 \RequirePackage{nccfoots}
3
```

```
\newcommand{\extrafootnoterule}{}

\let\defaultfootnoterule\footnoterule

\newcommand*{\SelectFootnoteRule}[2][0]{}

\newcommand{\footnoterulepriority}{1}

\newcommand{\SetFootnoteHook}[1]{}
\@onlypreamble\SetFootnoteHook

\newcommand{\SplitNote}{}

\newcommand*\ExtraParaSkip[1]{}

\newcommand*{\newfootnote}[2][plain]{%
    \ifstrequal{#2}{default}{}{% not "default"
        \expandafter\newbox\csname LWR@footnote#2box\endcsname%
        \appto{\LWR@printpendingfootnotes}{%
            \LWR@@printpendingfootnotes{footnote#2}%
        }
        \long\csdef{Footnotetext#2}##1##2{%
            \NCC@makefnmark{##1}%
            \LWR@@footnotetext{##2}{LWR@footnote#2box}%
        }%
        \long\csdef{Footnotetext#2+}##1##2{%
            \NCC@makefnmark{##1}%
            \LWR@@footnotetext{##2}{LWR@footnote#2box}%
        }%
    }% not "default"
}
\@onlypreamble\newfootnote

\newcommand*{\DeclareNewFootnote}[2][plain]{%
  \@ifnextchar[%
        {\LWR@manyfoot@declare{#1}{#2}}%
        {\LWR@manyfoot@declare{#1}{#2}[arabic]}%
}

\def\LWR@manyfoot@declare#1#2[#3]{%
\ifstrequal{#2}{default}{}{% not "default"
  \newfootnote[#1]{#2}%
  \newcounter{footnote#2}%
    \newcounter{footnote#2Reset}%
    \setcounter{footnote#2Reset}{0}%
    \csdef{thefootnote#2}{%
      \expandafter\noexpand\csname @#3\endcsname%
      \expandafter\noexpand\csname c@footnote#2\endcsname%
    }%
```

For bigfoot, the footnote commands may be appended with one or two plusses or one or two minuses, which are ignored in HTML.

```
  \expandafter\NewDocumentCommand\csname footnote#2\endcsname{t{+}t{+}t{-}t{-}}{%
        \stepcounter{footnote#2}%
        \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
        \@footnotemark%
        \csuse{Footnotetext#2}{\@thefnmark}% absorbs the footnote contents
    }%
    \csdef{footnotemark#2}{%
```

```
60          \stepcounter{footnote#2}%
61          \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
62          \@footnotemark%
63      }%
64    \expandafter\NewDocumentCommand\csname footnotetext#2\endcsname{t{+}t{+}t{-}t{-}}{%
65          \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
66          \csuse{Footnotetext#2}{\@thefnmark}% absorbs the footnote contents
67      }%
68      \csdef{Footnotemark#2}{%
69        \Footnotemark%
70      }%
71      \csdef{Footnote#2}##1{%
72        \Footnotemark{##1}%
73        \csuse{Footnotetext#2}{##1}%
74      }%
75  }% not "default"
76  }
77  \@onlypreamble\DeclareNewFootnote
```

File 276 **lwarp-marginal.sty**

§ 385 Package marginal

`marginal` (*Pkg*) marginal is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{marginal}
```

```
2 \newcommand*{\showlostmarginals}{}
3 \newcommand*{\enlargefreelist}{}
4 \newcommand*{\onesidemarginals}{}
```

File 277 **lwarp-marginfit.sty**

§ 386 Package marginfit

`marginfit` (*Pkg*) marginfit is ignored.

for HTML output: Discard all options for lwarp-marginfit:

```
1 \LWR@ProvidesPackageDrop{marginfit}[2018/06/08]
```

File 278 **lwarp-marginfix.sty**

§ 387 Package marginfix

(*Emulates or patches code by* Stephen Hicks.)

`marginfix` (*Pkg*) marginfix is ignored.

for HTML output: Discard all options for lwarp-marginfix:

```
1 \LWR@ProvidesPackageDrop{marginfix}[2013/09/08]
```

```
\newcommand*{\marginskip}[1]{}
\newcommand*{\clearmargin}{}
\newcommand*{\softclearmargin}{}
\newcommand*{\extendmargin}[1]{}
\newcommand*{\mparshift}[1]{}
\newdimen\marginheightadjustment
\newdimen\marginposadjustment
\newcommand*{\blockmargin}[1][]{}
\newcommand*{\unblockmargin}[1][]{}
\newcommand*{\marginphantom}[2][]{}
```

File 279 **lwarp-marginnote.sty**

§ 388 Package marginnote

(*Emulates or patches code by* Markus Kohm.)

marginnote (*Pkg*) marginnote is emulated.

for HTML output: Discard all options for lwarp-marginnote:

```
\LWR@ProvidesPackageDrop{marginnote}[2018/08/09]
```

```
\NewDocumentCommand{\marginnote}{+o +m o}{\marginpar{#2}}
```

```
\newcommand*{\marginnoteleftadjust}{}
\newcommand*{\marginnoterightadjust}{}
\newcommand*{\marginnotetextwidth}{}
\let\marginnotetextwidth\textwidth
\newcommand*{\marginnotevadjust}{}
\newcommand*{\marginfont}{}
\newcommand*{\raggedleftmarginnote}{}
\newcommand*{\raggedrightmarginnote}{}
```

```
\appto\LWR@restoreorigformatting{%
    \RenewDocumentCommand{\marginnote}{+o +m o}{}
}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\LWRmarginnote}[1][]{}}
\CustomizeMathJax{\newcommand{\marginnote}[2][]{\qquad{\small\textrm{#2}}\LWRmarginnote}}
\end{warpMathJax}
```

File 280 **lwarp-marvosym.sty**

§ 389 Package marvosym

(*Emulates or patches code by* Thomas Henlich, Mojca Miklavec.)

marvosym (*Pkg*) marvosym is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

for HTML output:

```
\LWR@ProvidesPackagePass{marvosym}[2011/07/20]

\renewcommand{\mvchr}[1]{%
    \begin{lateximage}*[symbol #1][marvosym #1]%
    \mvs\char#1%
    \end{lateximage}%
}

\renewcommand{\textmvs}[1]{%
    \begin{lateximage}%
    \mvs #1%
    \end{lateximage}%
}
```

File 281 **lwarp-mathalpha.sty**

§ 390 Package mathalpha

(*Emulates or patches code by* Michael Sharpe.)

`mathalpha` (*Pkg*) mathalpha is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options, and some bold fonts may not be not supported by MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mathalpha}[2021/11/18]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mathbfbb}[1]{\boldsymbol{\mathbb{#1}}}}% not bold
\CustomizeMathJax{\newcommand{\mathbfcal}[1]{\boldsymbol{\mathcal{#1}}}}
\CustomizeMathJax{\newcommand{\mathbffrak}[1]{\boldsymbol{\mathfrak{#1}}}}
\CustomizeMathJax{\newcommand{\mathbfscr}[1]{\boldsymbol{\mathscr{#1}}}}% not bold

\IfPackageLoadedWithOptionsTF{mathalpha}{oldbold}
{
\CustomizeMathJax{\newcommand{\mathbbb}[1]{\boldsymbol{\mathbb{#1}}}}% not bold
\CustomizeMathJax{\newcommand{\mathbcal}[1]{\boldsymbol{\mathcal{#1}}}}
\CustomizeMathJax{\newcommand{\mathbfrak}[1]{\boldsymbol{\mathfrak{#1}}}}
\CustomizeMathJax{\newcommand{\mathbscr}[1]{\boldsymbol{\mathscr{#1}}}}% not bold
}{}
\end{warpMathJax}
```

File 282 **lwarp-mathastext.sty**

§ 391 Package mathastext

(*Emulates or patches code by* Jean-François Burnol.)

`mathastext` (*Pkg*) mathastext is used as-is for SVG math, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mathastext}[2019/11/16]
```

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\ifmst@itgreek
%     \LWR@mathjax@addgreek@l@it{}{}
\else
    \ifmst@upgreek
        \LWR@mathjax@addgreek@l@up{}{}
    \else
        \ifmst@frenchmath
            \LWR@mathjax@addgreek@l@up{}{}
        \else
            \ifmst@italic
%               \LWR@mathjax@addgreek@l@it{}{}
            \else
                \LWR@mathjax@addgreek@l@up{}{}
            \fi
        \fi
    \fi
\fi

\ifcase\mst@greek@select
    \or{\LWR@mathjax@addgreek@u@it*{}{}}
%     \or{\LWR@mathjax@addgreek@u@up*{}{}}
\fi

\CustomizeMathJax{\newcommand{\mathnormalbold}[1]{\boldsymbol{#1}}}
\CustomizeMathJax{\newcommand{\MathEulerBold}[1]{\boldsymbol{#1}}}
\CustomizeMathJax{\newcommand{\MathEuler}[1]{{#1}}}
\CustomizeMathJax{\newcommand{\MathPSymbol}[1]{{#1}}}
\CustomizeMathJax{\let\fouriervec\vec}
\CustomizeMathJax{\let\pmvec\vec}
\CustomizeMathJax{\let\inodot\imath}
\CustomizeMathJax{\let\jnodot\jmath}
\CustomizeMathJax{\let\shortiff\iff}
\CustomizeMathJax{\let\longto\longrightarrow}
\CustomizeMathJax{\newcommand{\inftypsy}{\mathord{\unicode{x221E}}}}
\CustomizeMathJax{\newcommand{\proptopsy}{\mathrel{\unicode{x221D}}}}
\CustomizeMathJax{\let\prodpsy\prod}
\CustomizeMathJax{\let\sumpsy\sum}
\CustomizeMathJax{\let\MToriginalprod\prod}
\CustomizeMathJax{\let\MToriginalsum\sum}
\CustomizeMathJax{\newcommand{\DotTriangle}{\mathord{\unicode{x2234}}}}
\end{warpMathJax}
```

File 283 **lwarp-mathcomp.sty**

§ 392 Package mathcomp

(*Emulates or patches code by* Tilmann Böß.)

mathcomp (*Pkg*) mathcomp is supported as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mathcomp}[2001/01/07]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\tcohm}{\mathrm{\Omega}}}
```

```
\CustomizeMathJax{\newcommand{\tccelsius}{\unicode{x2103}}}
\CustomizeMathJax{\newcommand{\tcmu}{\mathrm{\unicode{x00B5}}}}
\CustomizeMathJax{\newcommand{\tcperthousand}{\unicode{x2030}}}
\CustomizeMathJax{\newcommand{\tcpertenthousand}{\unicode{x2031}}}
\CustomizeMathJax{\newcommand{\tcdegree}{\mathrm{^\circ}}}
\CustomizeMathJax{\newcommand{\tcdigitoldstyle}[1]{\oldstyle{#1}}}
\end{warpMathJax}
```

File 284 **lwarp-mathdesign.sty**

§ 393 Package

mathdesign

(*Emulates or patches code by* Paul Pichaureau.)

mathdesign (*Pkg*) mathdesign is used as-is for svg math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options except greekuppercase and greeklowercase. The dedicated macros for upright and italic greek letters work correctly, although the user may wish to swap the definitions for epsilon and phi.

svg math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{mathdesign}[2013/08/29]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{mathdesign}
```

Default greek upright or italicized:

```
\if@MD@grupright
\LWR@mathjax@addgreek@l@up{}{}
\fi

\if@MD@GRupright
\else
\LWR@mathjax@addgreek@u@it*{}{}
\fi
```

Upright:

```
\LWR@mathjax@addgreek@l@up{}{up}
\LWR@mathjax@addgreek@u@up*{}{up}
```

Italicized:

```
\LWR@mathjax@addgreek@l@it{}{it}
\LWR@mathjax@addgreek@u@it*{}{it}
```

Adapt to mathdesign inconsistency:

```
\CustomizeMathJax{\let\digammaup\Digammaup}
\CustomizeMathJax{\renewcommand{\digammait}{\mathit{\digammaup}}}
```

Extra symbols:

```
\CustomizeMathJax{\newcommand{\smallin}{\mathrel{\unicode{x220A}}}}
\CustomizeMathJax{\newcommand{\smallowns}{\mathrel{\unicode{x220D}}}}
\CustomizeMathJax{\newcommand{\notsmallin}{\mathrel{\LWRoverlaysymbols{/}{\unicode{x220A}}}}}
\CustomizeMathJax{\newcommand{\notsmallowns}{\mathrel{\LWRoverlaysymbols{/}{\unicode{x220D}}}}}
\CustomizeMathJax{\newcommand{\rightangle}{\mathord{\unicode{x221F}}}}
```

Integrals:

```
\CustomizeMathJax{\newcommand{\intclockwise}{\mathop{\unicode{x2231}}\limits}}
\CustomizeMathJax{\newcommand{\ointclockwise}{\mathop{\unicode{x2232}}\limits}}
\CustomizeMathJax{\newcommand{\ointctrclockwise}{\mathop{\unicode{x2233}}\limits}}
\CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
```

Math and text mode:

```
\CustomizeMathJax{\newcommand{\ddag}{\unicode{x2021}}}
\CustomizeMathJax{\newcommand{\P}{\unicode{x00B6}}}
\CustomizeMathJax{\newcommand{\copyright}{\unicode{x00A9}}}
\CustomizeMathJax{\newcommand{\dag}{\unicode{x2020}}}
\CustomizeMathJax{\newcommand{\pounds}{\unicode{x00A3}}}
```

Extra symbols:

```
\CustomizeMathJax{\newcommand{\iddots}{\mathinner{\unicode{x22F0}}}}
\CustomizeMathJax{\newcommand{\utimes}{\mathbin{\overline{\times}}}}
\CustomizeMathJax{\newcommand{\dtimes}{\mathbin{\underline{\times}}}}
\CustomizeMathJax{\newcommand{\udtimes}{\mathbin{\overline{\underline{\times}}}}}
\CustomizeMathJax{\newcommand{\leftwave}{\left\{}}
\CustomizeMathJax{\newcommand{\rightwave}{\right\}}}

\end{warpMathJax}
```

File 285 **lwarp-mathdots.sty**

§ 394 Package

mathdots

(*Emulates or patches code by* Dan Luecking.)

mathdots (*Pkg*) mathdots is used as-is for SVG math, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mathdots}[2014/06/11]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\iddots}{\mathinner{\unicode{x22F0}}}}
\CustomizeMathJax{\let\fixedddots\ddots}
\CustomizeMathJax{\let\fixedvdots\vdots}
\CustomizeMathJax{\let\fixediddots\iddots}
\CustomizeMathJax{\let\originalddots\ddots}
\CustomizeMathJax{\let\originalvdots\vdots}
\CustomizeMathJax{\let\originaliddots\iddots}
```

```
\CustomizeMathJax{\let\originaldddot\dddot}
\CustomizeMathJax{\let\originalddddot\ddddot}
\end{warpMathJax}
```

File 286 **lwarp-mathfixs.sty**

§ 395 Package mathfixs

(*Emulates or patches code by* Niklas Beisert.)

mathfixs (*Pkg*) mathfixs is used as-is for SVG math, and is emulated for MathJax.

⚠ Greek letters are unchanged.

for HTML output:

```
\LWR@ProvidesPackagePass{mathfixs}[2018/12/30]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\rfrac}[2]{\tfrac{#1}{#2}}}
\CustomizeMathJax{\newcommand{\vfrac}[2]{\mathinner{{}^{#1}\!/{}_{#2}}}}
\CustomizeMathJax{\newcommand{\ProvideMathFix}[1]{}}
\CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
\CustomizeMathJax{\newcommand{\.}{\,}}
\end{warpMathJax}
```

File 287 **lwarp-mathpazo.sty**

§ 396 Package mathpazo

(*Emulates or patches code by* Walter Schmidt.)

mathpazo (*Pkg*) mathpazo is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options. The dedicated macros for upright greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{mathpazo}[2020/03/25]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{mathpazo}

\ifpazo@slGreek
\LWR@mathjax@addgreek@u@it*{}{}
\fi

\LWR@mathjax@addgreek@u@up*{up}{}

\CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
\end{warpMathJax}
```

File 288 **lwarp-mathptmx.sty**

§ 397 Package mathptmx

(*Emulates or patches code by* Walter Schmidt.)

mathptmx (*Pkg*) mathptmx is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options. The dedicated macros for upright greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{mathptmx}[2020/03/25]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{mathptmx}

\IfPackageLoadedWithOptionsTF{mathptmx}{slantedGreek}
    {\LWR@mathjax@addgreek@u@it*{}{}}
    {}

\LWR@mathjax@addgreek@u@up*{up}{}
\end{warpMathJax}
```

File 289 **lwarp-mathspec.sty**

§ 398 Package mathspec

(*Emulates or patches code by* Andrew Gilbert Moschou.)

mathspec (*Pkg*) mathspec is used as-is with SVG math, and is emulated for MathJax.

⚠ quotes Double quotes (\" and the " character) are removed during MathJax emulation, but this also includes inside \text.

for HTML output:

```
\LWR@ProvidesPackagePass{mathspec}[2016/12/22]
```

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
```

Neutralize double quotes (" and \"):

```
\booltrue{LWR@MathJax@silentquotes}
```

Sort options for out Greek emulation:

```
\AtBeginDocument{
\ifcase\eu@GreekUppercase@@value %% If Greek Uppercase Regular
    \LWR@mathjax@addgreek@u@up*{}{}
\or %% If Greek Uppercase Italic
    \LWR@mathjax@addgreek@u@it*{}{}
\or %% If Greek Uppercase Plain
    \LWR@mathjax@addgreek@u@up*{}{}
\fi
\ifcase\eu@GreekLowercase@@value %% If Greek Lowercase Regular
    \LWR@mathjax@addgreek@l@up{}{}
\or %% If Greek Lowercase Italic
    \LWR@mathjax@addgreek@l@it{}{}
\or %% If Greek Lowercase Plain
    \LWR@mathjax@addgreek@l@it{}{}
\fi
}
```

Swap definitions according the mathspec conditionals:

```
\newcommand*{\LWR@mathspec@varforms}{%
\eu@ifbooltrue{GreekLowercase}{
    \eu@ifbooltrue{exchangebetaforms}{
        \CustomizeMathJax{\let\LWRorigbeta\beta}
        \CustomizeMathJax{\let\beta\varbeta}
        \CustomizeMathJax{\let\varbeta\LWRorigbeta}
    }
    \eu@ifbooltrue{exchangeepsilonforms}{
        \CustomizeMathJax{\let\LWRorigepsilon\epsilon}
        \CustomizeMathJax{\let\epsilon\varepsilon}
        \CustomizeMathJax{\let\varepsilon\LWRorigepsilon}
    }
    \eu@ifbooltrue{exchangethetaforms}{
        \CustomizeMathJax{\let\LWRorigtheta\theta}
        \CustomizeMathJax{\let\theta\vartheta}
        \CustomizeMathJax{\let\vartheta\LWRorigtheta}
    }
    \eu@ifbooltrue{exchangekappaforms}{
        \CustomizeMathJax{\let\LWRorigkappa\kappa}
        \CustomizeMathJax{\let\kappa\varkappa}
        \CustomizeMathJax{\let\varkappa\LWRorigkappa}
    }
    \eu@ifbooltrue{exchangepiforms}{
        \CustomizeMathJax{\let\LWRorigpi\pi}
        \CustomizeMathJax{\let\pi\varpi}
        \CustomizeMathJax{\let\varpi\LWRorigpi}
    }
    \eu@ifbooltrue{exchangerhoforms}{
        \CustomizeMathJax{\let\LWRorigrho\rho}
        \CustomizeMathJax{\let\rho\varrho}
        \CustomizeMathJax{\let\varrho\LWRorigrho}
    }
    \eu@ifbooltrue{exchangephiforms}{
        \CustomizeMathJax{\let\LWRorigphi\phi}
        \CustomizeMathJax{\let\phi\varphi}
        \CustomizeMathJax{\let\varphi\LWRorigphi}
    }
}
\eu@ifbooltrue{GreekUppercase}{
    \eu@ifbooltrue{exhangeThetaforms}{
        \CustomizeMathJax{\let\LWRorigTheta\Theta}
```

```
        \CustomizeMathJax{\let\Theta\varTheta}
        \CustomizeMathJax{\let\varTheta\LWRorigTheta}
    }
}
}
```

Append new action to mathspec's `\AtBeginDocument` code:

```
\xapptocmd{\exchangeforms}
    {\AtBeginDocument{\LWR@mathspec@varforms}}
    {}
    {\LWR@patcherror{mathspec}{exchangeforms}}

\end{warpMathJax}
```

File 290 **lwarp-mathtools.sty**

§ 399 Package

mathtools

(*Emulates or patches code by* Morten Høgholm, Lars Madsen.)

`mathtools` (*Pkg*)

⚠ equation numbering

mathtools is patched for use by lwarp. Emulation macros are provided for MathJax.

`showonlyrefs` is disabled, as it conflicts with cleveref, which is used by lwarp. Equation numbers may not match the print version.

⚠ italic correction

`mathic` is not emulated for HTML.

⚠ MathJax

If using MathJax:

- Recent changes may not yet be updated in the MathJax extension, which is used by lwarp.
- mathtools `disallowspaces` does not work for MathJax. Protect brackets which are not optional arguments, such as:

  ```
  \begin{gathered}{}
  [p]=1 . . .
  \end{gathered}
  ```

- `showonlyrefs` does not work in MathJax, and will result in a difference in equation numbering compared to the print version.
- `aligned` in MathJax requires math mode, but in LaTeX it doesn't. It may be required to use `warpHTML` and `warpprint` to isolate a version for each mode.
- `\DeclarePairedDelimiter` and related must be in the preamble before `\begin{document}`.

for HTML output:

```
\LWR@ProvidesPackagePass{mathtools}[2018/01/08]

\RequirePackage{graphicx}

\MHInternalSyntaxOn
```

Forces showonlyrefs off because lwarp uses cleveref, which is not compatible with showonlyrefs.

```
\renewcommand*\MT_showonlyrefs_true:{%
    \PackageWarningNoLine{lwarp}
    {%
        Mathtools \space showonlyrefs \space conflicts \space
        with \space cleveref,\MessageBreak
        which \space is \space used \space by \space lwarp, \space
        so \space showonlyrefs \space is\MessageBreak
        forced \space off. \space\space
        Equation \space numbers \space may \space not \space match%
    }
    \MT_showonlyrefs_false:
}
\mathtoolsset{showonlyrefs=false}
```

Forces math italic correction off. Not patched for lwarp.

```
\renewcommand*{\MT_mathic_true:}{\MT_mathic_false:}
\mathtoolsset{mathic=false}

\MHInternalSyntaxOff
```

For MathJax.

The MathJax package is used, and improvements are added.

```
\begin{warpMathJax}
\CustomizeMathJax{\require{mathtools}}

\LWR@infoprocessingmathjax{mathtools}

\CustomizeMathJax{\newenvironment{crampedsubarray}[1]{}{}}

\CustomizeMathJax{\newcommand{\smashoperator}[2][]{#2\limits}}

\CustomizeMathJax{\newcommand{\SwapAboveDisplaySkip}{}}

\CustomizeMathJax{\newcommand{\LaTeXunderbrace}[1]{\underbrace{#1}}}
\CustomizeMathJax{\newcommand{\LaTeXoverbrace}[1]{\overbrace{#1}}}


\CustomizeMathJax{\newcommand{\LWRmultlined}[1][]{\begin{multline*}}}
\CustomizeMathJax{\newenvironment{multlined}[1][]{\LWRmultlined}{\end{multline*}}}

\CustomizeMathJax{\let\LWRorigshoveleft\shoveleft}
\CustomizeMathJax{\renewcommand{\shoveleft}[1][]{\LWRorigshoveleft}}
\CustomizeMathJax{\let\LWRorigshoveright\shoveright}
\CustomizeMathJax{\renewcommand{\shoveright}[1][]{\LWRorigshoveright}}

\CustomizeMathJax{\newcommand{\shortintertext}[1]{\text{#1}\notag \\}}

\LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiter\DeclarePairedDelimiter
\renewcommand{\DeclarePairedDelimiter}[3]{
    \LWR@mathtools@orig@DeclarePairedDelimiter{#1}{#2}{#3}
% starred:
    \appto\LWR@customizedMathJax{\LWRbackslash()}
    \appto\LWR@customizedMathJax{%
```

```
      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubstar\}%
    }%
    \appto\LWR@customizedMathJax{[2][]}%
    \appto\LWR@customizedMathJax{\{\{}%
    \LWR@subcustomizedmathjax{##1\left#2##2##1\right#3}%
    \appto\LWR@customizedMathJax{\}\}}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
% not starred:
    \appto\LWR@customizedMathJax{\LWRbackslash(}
    \appto\LWR@customizedMathJax{%
      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubnostar\}%
    }%
    \appto\LWR@customizedMathJax{[2][]}%
    \appto\LWR@customizedMathJax{\{\{}%
    \LWR@subcustomizedmathjax{##1#2##2##1#3}%
    \appto\LWR@customizedMathJax{\}\}}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
% user macro:
    \appto\LWR@customizedMathJax{\LWRbackslash(}
    \appto\LWR@customizedMathJax{%
        \LWRbackslash{}newcommand\{\LWRbackslash{}\macrotocsname{#1}\}%
        \{\LWRbackslash{}ifstar%
            \LWRbackslash{}\macrotocsname{#1}LWRsubstar%
            \LWRbackslash{}\macrotocsname{#1}LWRsubnostar%
        \}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
}
\@onlypreamble\DeclareParedDelimiter

% (DeclarePairedDelimiterX is already defined to use \DeclarePairedDelimiterXPP.)

\LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiterXPP\DeclarePairedDelimiterXPP
\DeclareDocumentCommand{\DeclarePairedDelimiterXPP}{m O{1} m m m m}{
   \LWR@mathtools@orig@DeclarePairedDelimiterXPP{#1}[#2]{#3}{#4}{#5}{#6}{#7}
% subsubstar, second opt arg
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \appto\LWR@customizedMathJax{%
      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubsubstar\}%
    }%
    \appto\LWR@customizedMathJax{[#2]}%
    \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}left}%
    \LWR@subcustomizedmathjax{#3#4#7}%
    \appto\LWR@customizedMathJax{\LWRbackslash{}right}%
    \LWR@subcustomizedmathjax{#5#6}%
    \appto\LWR@customizedMathJax{\}\}}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
% substar, first opt arg
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \appto\LWR@customizedMathJax{%
      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubstar\}[1][]%
    }%
    \appto\LWR@customizedMathJax{%
        \{
        \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
        \LWRbackslash\macrotocsname{#1}LWRsubsubstar
        \}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
% subsubnostar, second opt arg
```

```
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \appto\LWR@customizedMathJax{%
      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubsubnostar\}%
    }%
    \appto\LWR@customizedMathJax{[#2]}%
    \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}delimsize}%
    \LWR@subcustomizedmathjax{#3#4#7}%
    \appto\LWR@customizedMathJax{\LWRbackslash{}delimsize}%
    \LWR@subcustomizedmathjax{#5#6}%
    \appto\LWR@customizedMathJax{\}\}}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
% subnostar, first opt arg
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \appto\LWR@customizedMathJax{%
      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubnostar\}[1][]%
    }%
    \appto\LWR@customizedMathJax{%
        \{
        \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
        \LWRbackslash\macrotocsname{#1}LWRsubsubnostar
        \}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
% user macro:
    \appto\LWR@customizedMathJax{\LWRbackslash(}
    \appto\LWR@customizedMathJax{%
        \LWRbackslash{}newcommand\{%
            \LWRbackslash{}\macrotocsname{#1}%
        \}%
            \{\LWRbackslash{}ifstar%
                \LWRbackslash{}\macrotocsname{#1}LWRsubstar%
                \LWRbackslash{}\macrotocsname{#1}LWRsubnostar%
            \}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
}
\@onlypreamble\DeclareParedDelimiterXPP
\@onlypreamble\DeclareParedDelimiterX

\CustomizeMathJax{\newcommand{\vcentcolon}{\mathrel{\unicode{x2236}}}}

\LetLtxMacro\LWR@mathtools@orig@newgathered\newgathered
\renewcommand{\newgathered}[4]{%
    \LWR@mathtools@orig@newgathered{#1}{#2}{#3}{#4}%
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \LWR@subcustomizedmathjax{%
        \newenvironment{#1}{\begin{gathered}}{\end{gathered}}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)}%
}
\@onlypreamble\newgathered

\end{warpMathJax}
```

File 291 **lwarp-mattens.sty**

§ 400 Package

mattens

(*Emulates or patches code by* Danie Els.)

mattens (*Pkg*) mattens is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mattens}[2010/03/26]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\LWRmattensnull}{}}

\CustomizeMathJax{\newcommand{\LWRmattensnostar}[2][]{%
  {#1{\LWRmattensundercmd{\LWRmattensovercmd{\LWRmattenscross{\boldsymbol{#2}}}}}}%
}}

\CustomizeMathJax{\newcommand{\LWRmattensstar}[2][]{%
    {#1{\LWRmattensundercmd{\LWRmattensovercmd{\LWRmattenscross{#2}}}}}%
}}

\CustomizeMathJax{\newcommand{\LWRmattens}{
    \ifstar\LWRmattensstar\LWRmattensnostar%
}}

\CustomizeMathJax{\newcommand{\aS}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\overrightarrow%
    \let\LWRmattensundercmd\LWRmattensnull%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\Sa}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\underrightarrow%
    \let\LWRmattensundercmd\LWRmattensnull%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\bS}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\overline%
    \let\LWRmattensundercmd\LWRmattensnull%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\Sb}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\underline%
    \let\LWRmattensundercmd\LWRmattensnull%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\aSa}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\overrightarrow%
```

```
    \let\LWRmattensundercmd\underrightarrow%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\aSb}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\overrightarrow%
    \let\LWRmattensundercmd\underline%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\bSa}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\overline%
    \let\LWRmattensundercmd\underrightarrow%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\bSb}{%
    \let\LWRmattenscross\LWRmattensnull%
    \let\LWRmattensovercmd\overline%
    \let\LWRmattensundercmd\underline%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\aCSa}{%
    \let\LWRmattenscross\tilde%
    \let\LWRmattensovercmd\overrightarrow%
    \let\LWRmattensundercmd\underrightarrow%
    \LWRmattens%
}}

\CustomizeMathJax{\newcommand{\bCSb}{%
    \let\LWRmattenscross\tilde%
    \let\LWRmattensovercmd\overline%
    \let\LWRmattensundercmd\underline%
    \LWRmattens%
}}
\end{warpMathJax}
```

File 292 **lwarp-maybemath.sty**

§ 401 Package maybemath

(*Emulates or patches code by* Andy Buckley.)

maybemath (*Pkg*) maybemath is used as-is for SVG math, and is emulated for MathJax.

⚠ no effect MathJax is not able to detect the surrounding text font, so all maybemath macros are ignored.

for HTML output:

```
\LWR@ProvidesPackagePass{maybemath}[2005/2/22]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mayberm}[1]{{#1}}}
\CustomizeMathJax{\let\maybebm\mayberm}
\CustomizeMathJax{\let\maybeit\mayberm}
```

```
\CustomizeMathJax{\let\maybeitrm\mayberm}
\CustomizeMathJax{\let\maybeitsubscript\mayberm}
\CustomizeMathJax{\let\maybesf\mayberm}
\CustomizeMathJax{\let\maybebmsf\mayberm}
\end{warpMathJax}
```

File 293 **lwarp-mcaption.sty**

§ 402 Package mcaption

(*Emulates or patches code by* Stephan Hennig.)

mcaption (*Pkg*) mcaption is ignored.

for HTML output: Discard all options for lwarp-mcaption:

```
\LWR@ProvidesPackageDrop{mcaption}[2009/03/13]
```

```
\newenvironment{margincap}{}{}
\newcommand*{\margincapalign}{}
\newlength{\margincapsep}
```

File 294 **lwarp-mdframed.sty**

§ 403 Package mdframed

(*Emulates or patches code by* Marco Daniel, Elke Schubert.)

mdframed (*Pkg*) mdframed is loaded with options forced to `frame method=none`.

§ 403.1 Limitations

support Most basic functionality is supported, including frame background colors and single-border colors and thickness, title and subtitle background colors and borders and thickness, border radius, and shadow. CSS classes are created for mdframed environments and frame titles.

⚠ loading When used, lwarp loads mdframed in HTML with `frame method=none`.

font For title font, use

```
frametitlefont=\textbf,
```

instead of

```
frametitlefont=\bfseries,
```

where `\textbf` must appear just before the comma and will receive the following text as its argument (since the text happens to be between braces in the mdframed source). Since lwarp does not support `\bfseries` and friends, only one font selection may be made at a time.

theoremtitlefont `theoremtitlefont` is not supported, since the following text is not in braces in the mdframed source.

ignored options

userdefinedwidth and align are currently ignored.

css classes

Environments created or encapsulated by mdframed are enclosed in a of class mdframed, and also class md<environmentname> for new environments.

Frame titles are placed in a of class |mdframedtitle|. Subtitles are in a of class |mdframedsubtitle|, and likewise for subsubtitles.

Pre-existing hooks are used to patch extra functions before and after the frames.

§ 403.2 Package loading

for HTML output:

```
\RequirePackage{xcolor}% for \convertcolorspec

\LWR@ProvidesPackageDrop{mdframed}[2013/07/01]
```

Do not require Ti*k*Z or pstricks:

```
\LWR@origRequirePackage[framemethod=none]{mdframed}
```

§ 403.3 Patches

Patch to remove PDF formatting and add HTML tags:

```
\AtBeginDocument{
\def\mdf@trivlist#1{%
  \edef\mdf@temp{%
%     \topsep=\the\topsep\relax%
%     \partopsep=\the\partopsep\relax%
%     \parsep=\the\parsep\relax%
  }%
%   \setlength{\topsep}{#1}%
%   \topskip\z@%
%   \partopsep\z@%
%   \parsep\z@%
%   \@nmbrlistfalse%
%   \@trivlist%
%   \labelwidth\z@%
%   \leftmargin\z@%
%   \itemindent\z@%
  \let\@itemlabel\@empty%
  \def\makelabel##1{##1}%
%   \item\relax\mdf@temp\relax%
}

\renewcommand*{\endmdf@trivlist}{%
\LWR@traceinfo{endmdf@trivlist}%
% \endtrivlist%
\LWR@listend%
}
}% AtBeginDocument
```

§ 403.4 Initial setup

To handle css and paragraphs, patch code at start and end of environment and contents. `\LWR@print@raggedright` helps avoid hyphenation.

```
\mdfsetup{
startcode={\LWR@mdframedstart\LWR@print@raggedright},
endcode={\LWR@mdframedend},
startinnercode={\LWR@startpars\LWR@print@raggedright},
endinnercode={\LWR@stoppars},
}
```

§ 403.5 Color and length HTML conversion

`\LWR@mdfprintcolor` {⟨*mdfcolorkey*⟩}

Given the mdframed key, print the color.

```
\newcommand*{\LWR@mdfprintcolor}[1]{%
\convertcolorspec{named}{\@nameuse{mdf@#1}}{HTML}\LWR@tempcolor%
\LWR@origpound\LWR@tempcolor
}
```

`\LWR@mdfprintlength` {⟨*mdflengthkey*⟩}

Given the mdframed key, print the length.

```
\newcommand*{\LWR@mdfprintlength}[1]{%
\LWR@forceminwidth{\@nameuse{mdf@#1@length}}%
\LWR@printlength{\LWR@atleastonept}%
}
```

§ 403.6 Environment encapsulation

`\LWR@mdframedstart` Actions before an mdframe starts.

Encapsulate a frame inside a `<div>` of the desired `class`.

```
\newcommand*{\LWR@mdframedstart}{%
\LWR@traceinfo{LWR@mdframedstart start}%
```

Warn if starting a frame inside a `<span>`:

```
    \LWR@spanwarninvalid{mdframe}%
```

Turn off paragraph handling during the generation of the encapsulating tags:

```
\LWR@stoppars%
```

Open a `<div>` and with custom `class` and custom `style`. A `BlockClass` environment is not used because this `<div>` is created by the mdframed `startcode` and `endcode` settings, which do not properly nest the `<div>` inside the `mdframed` environment.

```
\LWR@htmltagc{div class=\textquotedbl%
mdframed%
\ifdefstring{\LWR@mdthisenv}{mdframed}{}{ \LWR@mdthisenv}%
\textquotedbl \LWR@orignewline
style=\textquotedbl\LWR@orignewline
```

Convert and print the background color:

```
background: \LWR@mdfprintcolor{backgroundcolor} ; \LWR@orignewline
```

Convert and print the border color and width:

```
border: \LWR@mdfprintlength{linewidth} solid
\LWR@mdfprintcolor{linecolor} ; \LWR@orignewline
```

Convert and print the border radius:

```
border-radius: \LWR@mdfprintlength{roundcorner} ; \LWR@orignewline
```

Convert and print the shadow:

```
\ifbool{mdf@shadow}{%
    box-shadow:
    \LWR@mdfprintlength{shadowsize}
    \LWR@mdfprintlength{shadowsize}
    \LWR@mdfprintlength{shadowsize}
    \LWR@mdfprintcolor{shadowcolor} ;
}
{box-shadow: none ;}
\LWR@orignewline

\textquotedbl}
% \LWR@htmldivclass{\LWR@mdthisenv}
```

mdframed environment may not work with the HTML versions of the following, so restore them to their originals while inside mdframed:

```
\let\hspace\LWR@print@hspace%
\renewcommand*{\rule}{\LWR@print@rule}
\LetLtxMacro\makebox\LWR@print@makebox%

\LWR@startpars%
\LWR@traceinfo{LWR@mdframedstart done}%
}
```

\LWR@mdframedend

Actions after an mdframe ends.

After closing the , globally restore to the default environment type:

```
\newcommand*{\LWR@mdframedend}{
\LWR@traceinfo{LWR@mdframedend start}%
```

Close the custom :

```
\LWR@htmldivclassend{\LWR@mdthisenv}
```

Reset future custom class to the default:

```
\gdef\LWR@mdthisenv{mdframed}
```

Resume paragraph handling:

```
\LWR@startpars%
\LWR@traceinfo{LWR@mdframedend done}%
}
```

§ 403.7 Mdframed environment

```
\renewenvironment{mdframed}[1][]{%
 \color@begingroup%
   \mdfsetup{userdefinedwidth=\linewidth,#1}%
   \mdf@startcode%
   \mdf@preenvsetting%
   \ifdefempty{\mdf@firstframetitle}{}%
           {\let\mdf@frametitlesave\mdf@frametitle%
            \let\mdf@frametitle\mdf@firstframetitle%
           }%
   \ifvmode\nointerlineskip\fi%
```

```
        \ifdefempty{\mdf@frametitle}{}%
            {\mdfframedtitleenv{\mdf@frametitle}%
%             \mdf@@frametitle@use%
            }%
   \mdf@trivlist{\mdf@skipabove@length}%%
   \mdf@settings%
%    \mdf@lrbox{\mdf@splitbox@one}%
%    \mdf@startinnercode%
  }%
  {%
%    \mdf@@ignorelastdescenders%
   \par%
%     \unskip\ifvmode\nointerlineskip\hrule \@height\z@ \@width\hsize\fi%%
   \ifmdf@footnoteinside%
      \def\mdf@reserveda{%
        \mdf@footnoteoutput%
%         \mdf@endinnercode%
%         \endmdf@lrbox%
%         \ifdefempty{\mdf@frametitle}{}%
%             {\mdfframedtitleenv{\mdf@frametitle}\mdf@@frametitle@use}%
%         \detected@mdf@put@frame
      }%
   \else%
      \def\mdf@reserveda{%
%         \mdf@endinnercode%
%         \endmdf@lrbox%
%         \ifdefempty{\mdf@frametitle}{}%
%             {\mdfframedtitleenv{\mdf@frametitle}\mdf@@frametitle@use}%
%         \detected@mdf@put@frame%
        \mdf@footnoteoutput%
        }%
   \fi%
   \mdf@reserveda%
  \aftergroup\endmdf@trivlist%
 \color@endgroup%
 \mdf@endcode%
}
```

\mdf@footnoteoutput

```
\renewrobustcmd*\mdf@footnoteoutput{%
    \LWR@printpendingmpfootnotes%
}
```

§ 403.8 Titles and subtitles

\mdfframedtitleenv {⟨*title*⟩}

Place the title inside a of class mdframedtitle:

```
\newlength{\LWR@titleroundcorner}

\renewrobustcmd\mdfframedtitleenv[1]{%
\LWR@traceinfo{LWR@mdframedtitleenv start}%
```

Open a with a custom class and custom style:

```
\begin{BlockClass}[%
```

Convert and print the title background color:

```
background:
\LWR@mdfprintcolor{frametitlebackgroundcolor}
```

```
; \LWR@orignewline
```

Convert and print the title rule:

```
\ifbool{mdf@frametitlerule}{%
    border-bottom:
    \LWR@mdfprintlength{frametitlerulewidth}
    solid
    \LWR@mdfprintcolor{frametitlerulecolor}
    ; \LWR@orignewline
}{}%
```

Finish the custom style and the opening tag:

```
]{mdframedtitle}%
```

Print the title inside the :

```
\mdf@frametitlefont{\LWR@textcurrentfont{#1}}%
```

Close the :

```
\end{BlockClass}%
\LWR@traceinfo{LWR@mdframedtitleenv end}%
}
```

\LWR@mdfsubtitlecommon {⟨*sub — or — subsub*⟩} [⟨*options*⟩] {⟨*title*⟩}

Common code for \LWR@mdfsubtitle and \LWR@mdfsubsubtitle.

Encapsulate the subtitle inside a of class mdframedsubtitle:

```
\NewDocumentCommand{\LWR@mdfsubtitlecommon}{m o m}
{% the following empty line is required

\LWR@traceinfo{LWR@mdframedsubtitlecommon start}%
```

Open a with a custom class and custom style:

```
\begin{BlockClass}[%
```

Convert and print the background color:

```
background:
\LWR@mdfprintcolor{#1titlebackgroundcolor}
; \LWR@orignewline
```

Convert and print the above line:

```
\ifbool{mdf@#1titleaboveline}{%
    border-top:
    \LWR@mdfprintlength{#1titleabovelinewidth}
    solid
    \LWR@mdfprintcolor{#1titleabovelinecolor}
    ; \LWR@orignewline
}{}%
```

Convert and print the below line:

```
\ifbool{mdf@#1titlebelowline}{%
    border-bottom:
    \LWR@mdfprintlength{#1titlebelowlinewidth}
    solid
    \LWR@mdfprintcolor{#1titlebelowlinecolor}
    ; \LWR@orignewline
}{}%
```

Finish the custom style and the opening tag:

```
]{mdframed#1title}%
```

Perform the original subtitle action:

```
\IfNoValueTF{#2}
{\@nameuse{LWR@origmdf#1title}{\csuse{mdf@#1titlefont}{\LWR@textcurrentfont{#3}}}}%
{\@nameuse{LWR@origmdf#1title}[#2]{\csuse{mdf@#1titlefont}{\LWR@textcurrentfont{#3}}}}%
```

Close the :

```
\end{BlockClass}%
\LWR@traceinfo{LWR@mdframedsubtitlecommon end}%
}
```

`\LWR@mdfsubtitle` `[`⟨*options*⟩`]` `{`⟨*title*⟩`}`

```
\newcommand*{\LWR@mdfsubtitle}{%
\LWR@mdfsubtitlecommon{sub}%
}
\let\mdfsubtitle\LWR@mdfsubtitle
```

`\LWR@mdfsubsubtitle` `[`⟨*options*⟩`]` `{`⟨*title*⟩`}`

```
\newcommand*{\LWR@mdfsubsubtitle}{%
\LWR@mdfsubtitlecommon{subsub}%
}
\let\mdfsubsubtitle\LWR@mdfsubsubtitle
```

§ 403.9 New environments

`\LWR@mdthisenv` Stores the environment of the frame about to be created:

```
\newcommand*{\LWR@mdthisenv}{mdframed}
```

`\newmdenv` `[`⟨*options*⟩`]` `{`⟨*env-name*⟩`}`

Modified from the original to remember the environment.

```
\renewrobustcmd*\newmdenv[2][]{%
\newenvironment{#2}%
{%
\mdfsetup{#1}%
\renewcommand*{\LWR@mdthisenv}{md#2}%
\begin{mdframed}%
}
{\end{mdframed}}%
}
```

`\surroundwithmdframed` `[`⟨*options*⟩`]` `{`⟨*environment*⟩`}`

Modified from the original to remember the environment.

```
\renewrobustcmd*{\surroundwithmdframed}[2][]{%
\BeforeBeginEnvironment{#2}{%
\renewcommand*{\LWR@mdthisenv}{md#2}%
\begin{mdframed}[#1]}%
\AfterEndEnvironment{#2}{\end{mdframed}}%
}
```

`\mdtheorem` `[`⟨*mdframed-options*⟩`]` `{`⟨*envname*⟩`}` `[`⟨*numberedlike*⟩`]` `{`⟨*caption*⟩`}` `[`⟨*within*⟩`]`

Modified from the original to remember the environment.

```
\DeclareDocumentCommand{\mdtheorem}{ O{} m o m o }%
{\ifcsdef{#2}%
  {\mdf@PackageWarning{Environment #2 already exits\MessageBreak}}%
  {%
   \IfNoValueTF {#3}%
    {%#3 not given -- number relationship
     \IfNoValueTF {#5}%
       {%#3+#5 not given
       \@definecounter{#2}%
       \expandafter\xdef\csname the#2\endcsname{\@thmcounter{#2}}%
       \newenvironment{#2}[1][]{%
         \refstepcounter{#2}%
         \ifstrempty{##1}%
           {\let\@temptitle\relax}%
           {%
            \def\@temptitle{\mdf@theoremseparator%
                            \mdf@theoremspace%
                            \mdf@theoremtitlefont%
                            \LWR@textcurrentfont{##1}}% lwarp
            \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
            }%
        \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
                                        \@temptitle}]}%
         {\end{mdframed}}%
       \newenvironment{#2*}[1][]{%
         \ifstrempty{##1}{\let\@temptitle\relax}{\def\@temptitle{:\ ##1}}%
         \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
         {\end{mdframed}}%
       }%
       {%#5 given -- reset counter
       \@definecounter{#2}\@newctr{#2}[#5]%
       \expandafter\xdef\csname the#2\endcsname{\@thmcounter{#2}}%
       \expandafter\xdef\csname the#2\endcsname{%
             \expandafter\noexpand\csname the#5\endcsname \@thmcountersep%
                 \@thmcounter{#2}}%
       \newenvironment{#2}[1][]{%
         \refstepcounter{#2}%
         \ifstrempty{##1}%
           {\let\@temptitle\relax}%
           {%
            \def\@temptitle{\mdf@theoremseparator%
                            \mdf@theoremspace%
                            \mdf@theoremtitlefont%
                            \LWR@textcurrentfont{##1}}% lwarp
            \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
            }
        \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
                                        \@temptitle}]}%
         {\end{mdframed}}%
       \newenvironment{#2*}[1][]{%
         \ifstrempty{##1}%
           {\let\@temptitle\relax}%
           {%
            \def\@temptitle{\mdf@theoremseparator%
                            \mdf@theoremspace%
                            \mdf@theoremtitlefont%
                            \LWR@textcurrentfont{##1}}% lwarp
            \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
            }%
         \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
```

```
          {\end{mdframed}}%
        }%
    }%
    {%#3 given -- number relationship
       \global\@namedef{the#2}{\@nameuse{the#3}}%
       \newenvironment{#2}[1][]{%
         \refstepcounter{#3}%
         \ifstrempty{##1}%
           {\let\@temptitle\relax}%
           {%
            \def\@temptitle{\mdf@theoremseparator%
                            \mdf@theoremspace%
                            \mdf@theoremtitlefont%
                            \LWR@textcurrentfont{##1}}% lwarp
            \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
            }
         \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
                                         \@temptitle}]}%
         {\end{mdframed}}%
       \newenvironment{#2*}[1][]{%
         \ifstrempty{##1}{\let\@temptitle\relax}{\def\@temptitle{:\ ##1}}%
         \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
         {\end{mdframed}}%
    }%
   \BeforeBeginEnvironment{#2}{\renewcommand*{\LWR@mdthisenv}{md#2}}% lwarp
  \BeforeBeginEnvironment{#2*}{\renewcommand*{\LWR@mdthisenv}{md#2}}% lwarp
  }%
}
```

\newmdtheoremenv `[`⟨*1: mdframed-options*⟩`]` {⟨*2: envname*⟩} `[`⟨*3: numberedlike*⟩`]` {⟨*4: caption*⟩} `[`⟨*5: within*⟩`]`

Modified from the original to remember the environment.

```
\DeclareDocumentCommand\newmdtheoremenv{O{} m o m o }{%
 \ifboolexpr{ test {\IfNoValueTF {#3}} and test {\IfNoValueTF {#5}} }%
    {\newtheorem{#2}{#4}}%
    {%
     \IfValueT{#3}{\newtheorem{#2}[#3]{#4}}%
     \IfValueT{#5}{\newtheorem{#2}{#4}[#5]}%
    }%
\BeforeBeginEnvironment{#2}{%
\renewcommand*{\LWR@mdthisenv}{md#2}%
\begin{mdframed}[#1]}%
\AfterEndEnvironment{#2}{%
\end{mdframed}}%
}
```

File 295 **lwarp-mdwmath.sty**

§ 404 Package mdwmath

(*Emulates or patches code by* Mark Wooding.)

mdwmath (*Pkg*) mdwmath is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mdwmath}[1996/04/11]
```

```
\begin{warpMathJax}
\CustomizeMathJax{\let\LWRmdwmathsqrt\sqrt}
\CustomizeMathJax{\renewcommand{\sqrt}{\ifstar\LWRmdwmathsqrt\LWRmdwmathsqrt}}
\CustomizeMathJax{\newcommand{\bitand}{\mathbin\&}}
\CustomizeMathJax{\def\bitor{\mathbin\mid}}
\CustomizeMathJax{\def\dblor{\mathbin{\mid\mid}}}
\CustomizeMathJax{\def\dbland{\mathbin{\mathrel\bitand\mathrel\bitand}}}
\end{warpMathJax}
```

File 296 **lwarp-media9.sty**

§ 405 Package

media9

media9 (*Pkg*) media9 is emulated.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

⚠ & in a URL Many special characters are converted to regular catcode 12 characters for use inside a URL. & is used in the flash variables fields, which are split with xparse \SplitList, which does not seem to work with a catcode 12 divider token, so & is

not converted to catcode 12, and will not work in a URL with media9. Using & in a URL in a flashvars field may also cause parsing problems with print output, as well.

for HTML output:

```
\LWR@ProvidesPackageDrop{media9}[2019/02/21]

\LWR@origRequirePackage{lwarp-common-multimedia}

\RequirePackage{xkeyval}
```

\addmediapath {⟨*path*⟩}

Supported.

```
\newcommand*{\LWR@medianine@path}{}

\newcommand*{\addmediapath}[1]{\appto\LWR@medianine@path{{#1}}}
```

The options and poster text are reused in several places.

```
\newcommand*{\LWR@medianine@postertext}{}
\newcommand*{\LWR@medianine@options}{}
```

Each addresource can generate a multimedia object.

```
\define@key{LWR@medianine}{addresource}{%
    \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]
        {\LWR@medianine@postertext}
        {#1}
}
```

Each flashvars source can generate a multimedia object.

```
\newcommand*{\LWR@medianine@flashvarsb}[1]{%
    \IfBeginWith{#1}{source=}{%
        \StrGobbleLeft{#1}{7}[\LWR@tempone]%
        \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]%
            {\LWR@medianine@postertext}%
            {\LWR@tempone}%
    }{}%
    \IfBeginWith{#1}{src=}{%
        \StrGobbleLeft{#1}{4}[\LWR@tempone]%
        \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]%
            {\LWR@medianine@postertext}%
            {\LWR@tempone}%
    }{}%
}

\NewDocumentCommand{\LWR@medianine@flashvars}{ >{\SplitList{&}} m }{%
    \ProcessList {#1}{\LWR@medianine@flashvarsb}%
}

\define@key{LWR@medianine}{flashvars}{%
    \LWR@medianine@flashvars{#1}%
}
```

\includemedia [⟨*options*⟩] {⟨*poster text*⟩} {⟨*file or url*⟩}

```
\newcommand*{\LWR@includemediab}[3][]{%
    \let\input@path\LWR@medianine@path%
    \renewcommand*{\LWR@medianine@options}{#1}%
    \renewcommand*{\LWR@medianine@postertext}{#2}%
    \setkeys*{LWR@medianine}{#1}%
    \IfBeginWith{#3}{http}{\LWR@multimedia[#1]{#2}{#3}}{%
    \IfBeginWith{#3}{HTTP}{\LWR@multimedia[#1]{#2}{#3}}{%
    \IfBeginWith{#3}{ftp}{\LWR@multimedia[#1]{#2}{#3}}{%
    \IfBeginWith{#3}{FTP}{\LWR@multimedia[#1]{#2}{#3}}{%
    }}}}%
    \endgroup%
}

\newrobustcmd*{\includemedia}{%
    \begingroup%
    \LWR@linkmediacatcodes%
    \LWR@includemediab%
}
```

\mediabutton `[`⟨*options*⟩`]` `{`⟨*text*⟩`}`

Ignored.

```
\newcommand*{\mediabutton}[2][]{}
```

File 297 **lwarp-memhfixc.sty**

§ 406 Package memhfixc

memhfixc (*Pkg*) memhfixc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{memhfixc}[2013/05/30]
```

File 298 **lwarp-menukeys.sty**

§ 407 Package menukeys

(*Emulates or patches code by* Tobias Weh.)

menukeys (*Pkg*) menukeys is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{menukeys}[2020/12/19]
```

Patch to use a `lateximage` whose `alt` text is the contents of this use of the macro. A hash on these contents allows the reuse of the image for each instance of the same contents.

```
\xpatchcmd{\tw@define@menu@macro@}
    {\@nameuse{tw@style@#4@pre}}
    {%
        \begin{lateximage}*[\detokenize{##2}]%
        \@nameuse{tw@style@#4@pre}%
    }
    {}
```

```
    {\LWR@patcherror{menukeys}{tw@define@menu@macro@}}

\xpatchcmd{\tw@define@menu@macro@}
    {\@nameuse{tw@style@#4@post}}
    {%
        \@nameuse{tw@style@#4@post}%
        \end{lateximage}%
    }
    {}
    {\LWR@patcherror{menukeys}{tw@define@menu@macro@ B}}
```

Patch the existing macros:

```
\renewmenumacro{\menu}[>]{menus}
\renewmenumacro{\directory}[/]{paths}
\renewmenumacro{\keys}[+]{roundedkeys}
```

File 299 **lwarp-metalogo.sty**

§ 408 Package metalogo

(*Emulates or patches code by* Andrew Gilbert Moschou.)

metalogo (*Pkg*) metalogo is used in print mode, and emulated in HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{metalogo}[2010/05/29]

\newcommand*{\LWR@HTML@setlogokern}[2]{}
\newcommand*{\LWR@HTML@setlogodrop}[2][XeTeX]{}
\newcommand*{\LWR@HTML@setLaTeXa}[1]{}
\newcommand*{\LWR@HTML@setLaTeXee}[1]{}
\newcommand*{\LWR@HTML@seteverylogo}[1]{}
\newcommand*{\LWR@HTML@everylogo}[1]{}

\LWR@formatted{setlogokern}
\LWR@formatted{setlogodrop}
\LWR@formatted{setLaTeXa}
\LWR@formatted{setLaTeXee}
\LWR@formatted{seteverylogo}
\LWR@formatted{everylogo}
```

File 300 **lwarp-metalogox.sty**

§ 409 Package metalogox

(*Emulates or patches code by* Brian Dunn.)

metalogox (*Pkg*) metalogox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{metalogox}[2019/01/20]
```

\AtBeginDocument, adjust the logo setting according to the font which is active at that moment.

```
\AtBeginDocument{
    \let\LWR@metalogox@currentformatting\LWR@formatting
    \renewcommand*{\LWR@formatting}{print}%
    \autoadjustlogos*
    \let\LWR@formatting\LWR@metalogox@currentformatting
}
```

File 301 **lwarp-mhchem.sty**

§ 410 Package

mhchem

(*Emulates or patches code by* Martin Hensel.)

mhchem (*Pkg*) mhchem is patched for use by lwarp.

without MathJax Without MathJax, mhchem expressions are converted to SVG math. Inline expressions use hashed filenames to allow reuse, and assume that any mhchem options are global.

MathJax with mhchem extension For MathJax, the mhchem extension is used if the mhchem expression is used inside a math expression:

```
$\ce{C6H5-CHO}$
```

To force the use of SVG math for an expression which does not work with MathJax, place the expression between `\displaymathother` and `\displaymathnormal`:

```
\displaymathother
\[ \ce{ . . . } \]     . . .     $ \ce { . . . } $
\displaymathnormal
```

not inside math If *not* used inside a math expression, lwarp converts standalone mhchem expressions into SVG math images.

⚠ nested math When producing HTML output without the MathJax mhchem extension, lwarp does not support the use of nested dollar signs in mhchem expressions.

For some examples from the mhchem manual, change as follows:

```
$\ce{NaOH(aq,$\infty$)}$            % old
$\ce{NaOH(aq,\infty)}$              % new

$\ce{Fe(CN)_{$\frac{6}{2}$}$}$      % old
$\ce{Fe(CN)_{\frac{6}{2}}}$         % new

$\ce{NO_$x$}$                       % old
$\ce{NO_x}$                         % new

$\ce{NO_${x}$}$                     % old
$\ce{NO_{x}}$                       % new

$\ce{$cis${-}[PtCl2(NH3)2]}$        % old
$\ce{\mathit{cis}{-}[PtCl2(NH3)2]}$ % new
```

for HTML output:

```
\LWR@ProvidesPackagePass{mhchem}[2018/06/22]
```

The original definition of \ce:

```
\LetLtxMacro\LWR@mhchem@origce\ce
```

The new definition, called from the new \ce after math shift is set. The starred lateximage uses a hashed filename for the SVG image. The alt tag is set to the mhchem expression.

```
\newcommand{\LWR@mhchem@HTML@ce}[1]{%
    \LWR@findcurrenttextcolor% sets \LWR@tempcolor
    \ifbool{LWR@xfakebold}%
        {\def\LWR@tempone{Y}}%
        {\def\LWR@tempone{N}}%
    \begin{lateximage}%
        *%
        [%
            \textbackslash{}%
            ce%
            \{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
        ]%
        *%
        [%
            FM\LWR@f@family%
            SR\LWR@f@series%
            SH\LWR@f@shape%
            SHC\LWR@f@shapecaps%
            CL\LWR@tempcolor%
            FB\LWR@tempone% xfakebold
        ]%
    \LWR@setcurrentfont%
    \LWR@mhchem@origce{#1}%
    \end{lateximage}%
    \endgroup%
    \addtocounter{LWR@mhchem@cedepth}{-1}%
}
```

Only set math shift if outer depth:

```
\newcounter{LWR@mhchem@cedepth}
\setcounter{LWR@mhchem@cedepth}{0}
```

The new \ce. Sets math shift then continues.

```
\renewcommand{\ce}{%
    \begingroup%
    \ifnumequal{\value{LWR@mhchem@cedepth}}{0}{%
        \catcode`\$=3% math shift
    }{}%
    \addtocounter{LWR@mhchem@cedepth}{1}%
    \LWR@mhchem@HTML@ce%
}
```

The original definition of \cesplit:

```
\LetLtxMacro\LWR@mhchem@origcesplit\cesplit
```

The new definition, called from the new \cesplit after math shift is set. The starred lateximage uses a hashed filename for the SVG image. The alt tag is set to the mhchem expression.

```
\newcommand*{\LWR@mhchem@HTML@cesplit}[2]
{%
    \LWR@findcurrenttextcolor% sets \LWR@tempcolor
    \ifbool{LWR@xfakebold}%
        {\def\LWR@tempone{Y}}%
        {\def\LWR@tempone{N}}%
    \begin{lateximage}%
        *%
        [%
            \textbackslash{}%
            cesplit%
            \{\LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}%
        ]%
        *%
        [%
            FM\LWR@f@family%
            SR\LWR@f@series%
            SH\LWR@f@shape%
            SHC\LWR@f@shapecaps%
            CL\LWR@tempcolor%
            FB\LWR@tempone% xfakebold
        ]%
    \LWR@setcurrentfont%
    \LWR@mhchem@origcesplit{#1}{#2}%
    \end{lateximage}%
    \endgroup%
}
```

Only set math shift if outer depth:

```
\newcounter{LWR@mhchem@cesplitdepth}
\setcounter{LWR@mhchem@cesplitdepth}{0}
```

The new \cesplit. Sets math shift then continues.

```
\renewcommand{\cesplit}{%
    \begingroup%
    \ifnumequal{\value{LWR@mhchem@cesplitdepth}}{0}{%
        \catcode`\$=3% math shift
    }{}%
    \addtocounter{LWR@mhchem@cesplitdepth}{1}%
    \LWR@mhchem@HTML@cesplit%
}
```

Resore originals inside a lateximage:

```
\appto\LWR@restoreorigformatting{%
\LetLtxMacro\ce\LWR@mhchem@origce%
\LetLtxMacro\cesplit\LWR@mhchem@origcesplit%
}

\begin{warpMathJax}
\CustomizeMathJax{\require{mhchem}}
\end{warpMathJax}
```

File 302 **lwarp-microtype.sty**

§ 411 Package microtype

(*Emulates or patches code by* R Schlicht.)

`microtype` (*Pkg*) microtype is pre-loaded by lwarp. All user options and macros are ignored and disabled.

for HTML output: Discard all options for lwarp-microtype:

```
\LWR@ProvidesPackageDrop{microtype}[2018/01/14]

\DeclareDocumentCommand{\DeclareMicrotypeSet}{o m m}{}
\DeclareDocumentCommand{\UseMicrotypeSet}{o m}{}
\DeclareDocumentCommand{\DeclareMicrotypeSetDefault}{o m}{}
\DeclareDocumentCommand{\SetProtrusion}{o m m}{}
\DeclareDocumentCommand{\SetExpansion}{o m m}{}
\DeclareDocumentCommand{\SetTracking}{o m m}{}
\DeclareDocumentCommand{\SetExtraKerning}{o m m}{}
\DeclareDocumentCommand{\SetExtraSpacing}{o m m}{}
\DeclareDocumentCommand{\DisableLigatures}{o m}{}
\DeclareDocumentCommand{\DeclareCharacterInheritance}{o m m}{}
\DeclareDocumentCommand{\DeclareMicrotypeVariants}{m}{}
\DeclareDocumentCommand{\DeclareMicrotypeAlias}{m m}{}
\DeclareDocumentCommand{\LoadMicrotypeFile}{m}{}
\DeclareDocumentCommand{\DeclareMicrotypeBabelHook}{m m}{}
\DeclareDocumentCommand{\microtypesetup}{m}{}
\DeclareDocumentCommand{\microtypecontext}{m}{}
\DeclareDocumentCommand{\textmicrotypecontext}{m m}{#2}
\IfPackageLoadedTF{letterspace}{\let\MT@textls\relax}{%
\DeclareDocumentCommand{\lsstyle}{}{}
\DeclareDocumentCommand{\textls}{o +m}{}
\DeclareDocumentCommand{\lslig}{m}{#1}
}
\def\DeclareMicrotypeSet#1#{\@gobbletwo}
\def\DeclareMicrotypeVariants#1#{\@gobble}
\@onlypreamble\DeclareMicrotypeSet
\@onlypreamble\UseMicrotypeSet
\@onlypreamble\DeclareMicrotypeSetDefault
\@onlypreamble\DisableLigatures
\@onlypreamble\DeclareMicrotypeVariants
\@onlypreamble\DeclareMicrotypeBabelHook
```

File 303 **lwarp-midfloat.sty**

§ 412 Package midfloat

(*Emulates or patches code by* Sigitas Tolušis.)

`midfloat` (*Pkg*) midfloat is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{midfloat}[2012/05/29]
```

```
\newenvironment{strip}[1][]{}{}
\newskip\stripsep
```

File 304 **lwarp-midpage.sty**

§ 413 Package midpage

midpage (*Pkg*) midpage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{midpage}[2009/09/03]
```

```
\newenvironment{midpage}
{\begin{BlockClass}[%
    \LWR@print@mbox{margin-top:6ex} ; \LWR@print@mbox{margin-bottom:6ex}%
]{midpage}}
{\end{BlockClass}}
```

File 305 **lwarp-minibox.sty**

§ 414 Package minibox

(*Emulates or patches code by* Will Robertson.)

minibox (*Pkg*) minibox is patched for use by lwarp.

Due to HTML limitations regarding paragraphs and s, miniboxes inline with other text will appear on their own line.

for HTML output:

```
\LWR@ProvidesPackagePass{minibox}[2013/06/21]
```

```
\ExplSyntaxOn
\newcommand\LWR@HTML@minibox[2][]{%
    \LWR@stoppars%
    \group_begin:
    \keys_set:nn {minibox} {#1}
    \bool_if:NTF \l_minibox_frame_bool
    {
        \setlength\fboxrule{\l_minibox_rule_dim}
        \setlength\fboxsep{\l_minibox_pad_dim}
        \fboxBlock{%
            \begin{tabular}[\l_minibox_tabular_valign_tl]%
              {\l_minibox_tabular_preamble_tl}
                {#2}
            \end{tabular}
        }%
    }
    {
        \begin{BlockClass}[display:inline-block]{minibox}
        \begin{tabular}[\l_minibox_tabular_valign_tl]%
          {\l_minibox_tabular_preamble_tl}
            {#2}
        \end{tabular}
        \end{BlockClass}
```

```
    }
    \group_end:
    \LWR@startpars%
}
\ExplSyntaxOff

\LWR@formatted{minibox}
```

File 306 **lwarp-minitoc.sty**

§ 415 Package minitoc

minitoc (*Pkg*) minitoc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{minitoc}[2018/07/12]
```

mtcoff disables minitoc.

```
\usepackage{mtcoff}
```

File 307 **lwarp-minted.sty**

§ 416 Package minted

(*Emulates or patches code by* Geoffrey M. Poore.)

minted (*Pkg*) minted is patched for use by lwarp.

⚠ limitations mathescape and highlightlines don't work. Line numbers on the right will not be aligned. Due to *pdftotext*, extra spaces may appear in broken lines if other formatting is included.

for HTML output:

```
\LWR@ProvidesPackagePass{minted}[2022/12/12]

\renewcommand{\setminted}[2][]{%
  \ifthenelse{\equal{#1}{}}%
    {\setkeys{minted@opt@g}{%
        #2,
      mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
    }}%
    {\minted@configlang{#1}%
      \setkeys{minted@opt@lang}{%
        #2,
      mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
    }}}

\renewcommand{\setmintedinline}[2][]{%
  \ifthenelse{\equal{#1}{}}%
    {\setkeys{minted@opt@g@i}{%
        #2,
      mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
    }}%
    {\minted@configlang{#1}%
      \setkeys{minted@opt@lang@i}{%
```

```
        #2,
      mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
    }}}

\xpatchcmd{\RobustMintInlineProcess}
    {\setkeys{minted@opt@cmd}{#1}}
    {%
        \setkeys{minted@opt@cmd}{%
            #1,%
            mathescape=false,breaklines,texcomments=false,highlightlines={}%
        }%
    }
    {}
    {\LWR@patcherror{minted}{minted}}

\xpatchcmd{\RobustMintProcess}
    {\setkeys{minted@opt@cmd}{#1}}
    {%
        \setkeys{minted@opt@cmd}{%
            #1,%
            mathescape=false,breaklines,texcomments=false,highlightlines={}%
        }%
    }
    {}
    {\LWR@patcherror{minted}{minted}}

\xpatchcmd{\minted}
    {\setkeys{minted@opt@cmd}{#1}}
    {%
        \setkeys{minted@opt@cmd}{%
            #1,%
            mathescape=false,breaklines,texcomments=false,highlightlines={}%
        }%
    }
    {}
    {\LWR@patcherror{minted}{minted}}

\xpatchcmd{\inputminted}
    {\setkeys{minted@opt@cmd}{#1}}
    {\setkeys{minted@opt@cmd}{%
            #1,%
            mathescape=false,breaklines,texcomments=false,highlightlines={}%
        }%
    }
    {}
    {\LWR@patcherror{minted}{inputminted}}



%\xpatchcmd{\mintinline}
%    {\setkeys{minted@opt@cmd}{#1}}
%    {\setkeys{minted@opt@cmd}{%
%            #1,%
%            mathescape=false,breaklines,texcomments=false,highlightlines={}%
%         }%
%    }
%    {}
%    {\LWR@patcherror{minted}{mintinline}}

%\xpatchcmd{\mint}
```

```
%    {\setkeys{minted@opt@cmd}{#1}}
%    {%
%        \setkeys{minted@opt@cmd}{%
%            #1,%
%          mathescape=false,breaklines,texcomments=false,highlightlines={}%
%        }%
%    }
%    {}
%    {\LWR@patcherror{minted}{mint}}

\renewenvironment{minted@snugshade*}[1]%
{%
    \colorlet{shadecolor}{#1}%
    \begin{snugshade*}%
}
{%
    \end{snugshade*}%
}
```

File 308 **lwarp-mismath.sty**

§ 417 Package

mismath

(*Emulates or patches code by* Antoine Missier.)

`mismath` (*Pkg*) mismath is patched for SVG math, and emulated for MathJax.

⚠ MathJax `\enumber`, `\inumber`, `\jnumber`, and `\pinumber` are ignored for MathJax, except that `\itpi` is made available as a clone of `\pi`.

`\MathUp`, `\MathIt`, `\MathNumbers`, and `\MathNormal` are ignored in MathJax.

For MathJax, `\boldvect` and `\arrowvect` are honored if in the preamble.

If `\boldvectcommand` is set to `\mathbf` in the preamble, it will be used for MathJax, otherwise it will default to `\mathit`. `\boldvectcommand` may also be set with `\CustomizeMathJax` in the preamble. See section 8.7.7. Note that as of this writing there is not a bold italic font across all MathJax fonts.

If `\probastyle` is set to `\mathbb` in the preamble, it will be used for MathJax, otherwise it will default to `\mathrm`. `\probastyle` may be set with `\CustomizeMathJax` in the preamble.

If `\mathset` is set to `\mathbb` in the preamble, it will be used for MathJax, otherwise it will default to `\mathbf`. `\mathset` may be set with `\CustomizeMathJax` in the preamble.

for HTML output:

```
\LWR@ProvidesPackagePass{mismath}[2023/02/24]
```

For MathJax, used in the HTML comment before the environment.

```
\ifbool{mathjax}{
    \RenewEnviron{mathcols}{%
        \preto\BODY{\begin{aligned}\displaystyle}
        \appto\BODY{\end{aligned}}
        \expandafter\(\BODY\)
    }
```

```
}% mathjax
```

For SVG math. The `lateximage` restores the original defintion of the `math` environment.

```
{% svg
    \renewenvironment{mathcols}{
        \begin{lateximage}
        \begin{math}
        \begin{aligned}\displaystyle
    }{
        \end{aligned}%
        \end{math}
        \end{lateximage}
    }
}% svg

\renewcommand{\changecol}{
    \end{aligned}   \qquad
    \begin{aligned}\displaystyle
}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mathup}[1]{\mathrm{#1}}}
\CustomizeMathJax{\newcommand{\e}{\mathrm{e}}}
\CustomizeMathJax{\newcommand{\i}{\mathrm{i}}}
\CustomizeMathJax{\newcommand{\j}{\mathrm{j}}}

\CustomizeMathJax{\let\mathbfsfit\mathbfit}% not sans
\CustomizeMathJax{\let\tensor\mathbfsfit}

\CustomizeMathJax{\newcommand{\boldvect}{}}
\CustomizeMathJax{\newcommand{\arrowvect}{}}
\CustomizeMathJax{\newcommand{\pinumber}[1][]{}}
\CustomizeMathJax{\newcommand{\hvect}[1]{\vec{\vphantom{h}#1}}}
\CustomizeMathJax{\newcommand{\hvec}[1]{\vec{\vphantom{t}#1}}}
\CustomizeMathJax{%
    \newcommand{\norm}[1]{\left\vert\left\vert#1\right\vert\right\vert}
}
\CustomizeMathJax{\newcommand{\di}{\mathop{}\!\mathrm{d}}}

\CustomizeMathJax{\newcommand{\P}{\operatorname{\probastyle{P}}}}
\CustomizeMathJax{\newcommand{\E}{\operatorname{\probastyle{E}}}}
\CustomizeMathJax{\newcommand{\V}{\operatorname{\probastyle{V}}}}
\CustomizeMathJax{\newcommand{\Par}{\unicode{x00B6}}}

\CustomizeMathJax{\DeclareMathOperator{\adj}{adj}}
\CustomizeMathJax{\DeclareMathOperator{\Aut}{Aut}}
\CustomizeMathJax{\DeclareMathOperator{\codim}{codim}}
\CustomizeMathJax{\DeclareMathOperator{\Conv}{Conv}}
\CustomizeMathJax{\DeclareMathOperator{\cov}{cov}}
\CustomizeMathJax{\DeclareMathOperator{\Cov}{Cov}}
\CustomizeMathJax{\newcommand{\curl}{\operatorname{\vect{\mathrm{curl}}}}}
\CustomizeMathJax{\DeclareMathOperator{\divg}{div}}
\CustomizeMathJax{\DeclareMathOperator{\End}{End}}

\CustomizeMathJax{\DeclareMathOperator{\erf}{erf}}
\CustomizeMathJax{\newcommand{\grad}{\operatorname{\vect{\mathrm{grad}}}}}
\CustomizeMathJax{\DeclareMathOperator{\id}{id}}
\CustomizeMathJax{\DeclareMathOperator{\Id}{Id}}
```

```
\CustomizeMathJax{\DeclareMathOperator{\im}{im}}
\CustomizeMathJax{\let\oldIm\Im}
\CustomizeMathJax{\renewcommand{\Im}{\operatorname{Im}}}
\CustomizeMathJax{\DeclareMathOperator{\lb}{lb}}
\CustomizeMathJax{\DeclareMathOperator{\lcm}{lcm}}

\CustomizeMathJax{\DeclareMathOperator{\rank}{rank}}
\CustomizeMathJax{\let\oldRe\Re}
\CustomizeMathJax{\renewcommand{\Re}{\operatorname{Re}}}
\CustomizeMathJax{\newcommand{\rot}{\operatorname{\vect{\mathrm{rot}}}}}
\CustomizeMathJax{\DeclareMathOperator{\sgn}{sgn}}
\CustomizeMathJax{\DeclareMathOperator{\sinc}{sinc}}
\CustomizeMathJax{\DeclareMathOperator{\spa}{span}}
\CustomizeMathJax{\DeclareMathOperator{\tr}{tr}}
\CustomizeMathJax{\DeclareMathOperator{\var}{var}}
\CustomizeMathJax{\DeclareMathOperator{\Var}{Var}}
\CustomizeMathJax{\DeclareMathOperator{\Zu}{Z}}

\CustomizeMathJax{\DeclareMathOperator{\arccot}{arccot}}
\CustomizeMathJax{\DeclareMathOperator{\sech}{sech}}
\CustomizeMathJax{\DeclareMathOperator{\csch}{csch}}
\CustomizeMathJax{\DeclareMathOperator{\arsinh}{arsinh}}
\CustomizeMathJax{\DeclareMathOperator{\arcosh}{arcosh}}
\CustomizeMathJax{\DeclareMathOperator{\artanh}{artanh}}
\CustomizeMathJax{\DeclareMathOperator{\arcoth}{arcoth}}
\CustomizeMathJax{\DeclareMathOperator{\arsech}{arsech}}
\CustomizeMathJax{\DeclareMathOperator{\arcsch}{arcsch}}

\CustomizeMathJax{\DeclareMathOperator{\bigO}{\mathcal{O}}}
\CustomizeMathJax{\DeclareMathOperator{\bigo}{O}}
\CustomizeMathJax{\DeclareMathOperator{\lito}{o}}

\CustomizeMathJax{\newcommand{\R}{\mathset{R}}}
\CustomizeMathJax{\newcommand{\C}{\mathset{C}}}
\CustomizeMathJax{\newcommand{\N}{\mathset{N}}}
\CustomizeMathJax{\newcommand{\Z}{\mathset{Z}}}
\CustomizeMathJax{\newcommand{\Q}{\mathset{Q}}}
\CustomizeMathJax{\newcommand{\F}{\mathset{F}}}
\CustomizeMathJax{\newcommand{\K}{\mathset{K}}}

\CustomizeMathJax{\newcommand{\ds}{\displaystyle}}
\CustomizeMathJax{\newcommand{\dlim}{\lim\limits}}
\CustomizeMathJax{\newcommand{\dsum}{\sum\limits}}
\CustomizeMathJax{\newcommand{\dprod}{\prod\limits}}
\CustomizeMathJax{\newcommand{\dcup}{\bigcup\limits}}
\CustomizeMathJax{\newcommand{\dcap}{\bigcap\limits}}
\CustomizeMathJax{\newcommand{\lbar}{\overline}}
\CustomizeMathJax{\newcommand{\hlbar}[1]{\overline{\vphantom{h}#1}}}
\CustomizeMathJax{\newcommand{\LWReqdefstar}{\stackrel{\Delta}{=}}}
\CustomizeMathJax{\newcommand{\LWReqdefnostar}{\stackrel{\mathrm{def}}{=}}}
\CustomizeMathJax{\newcommand{\eqdef}{\ifstar\LWReqdefstar\LWReqdefnostar}}
\CustomizeMathJax{\newcommand{\unbr}{\underbrace}}
\CustomizeMathJax{\newcommand{\iif}{if and only if }}

\CustomizeMathJax{\newcommand{\mul}{\mathord{\times}}}
\CustomizeMathJax{\newcommand{\then}{\ \Longrightarrow \ \mbox{} }}
\CustomizeMathJax{\newcommand{\txt}[1]{\quad\text{#1}\quad}}
\CustomizeMathJax{\newcommand{\pow}[2]{\left( #1 \right)^{\!#2}}}
\CustomizeMathJax{\newcommand{\abs}[1]{\left\vert#1\right\vert}}
\CustomizeMathJax{\newcommand{\lfrac}[2]{\frac{\:#1\:}{\:#2\:}}}
```

```
\CustomizeMathJax{\newenvironment{system}[1][l]%
    {\left\{\begin{array}{@{.15em}#1@{}}}
    {\end{array}\right.}
}

\CustomizeMathJax{\newenvironment{spmatrix}
    {\left(\begin{smallmatrix}}
    {\end{smallmatrix}\right)}
}

\CustomizeMathJax{%
    \newenvironment{mathcols}
        {\begin{aligned}\displaystyle}
        {\end{aligned}}
}
\CustomizeMathJax{\newcommand{\changecol}{\end{aligned}\qquad\begin{aligned}}}
```

User-adjustable settings, detected if in the preamble.

```
\AtBeginDocument{
\ifdef{\itpi}{
    \CustomizeMathJax{\let\itpi\pi}
}{}
\ifdefstring{\boldvectcommand}{\mathbf}{
    \CustomizeMathJax{\newcommand{\boldvectcommand}[1]{\mathbf{#1}}}
}{
    \CustomizeMathJax{\newcommand{\boldvectcommand}[1]{\boldsymbol{#1}}}
}
\ifbool{arrowvect}{
    \CustomizeMathJax{\newcommand{\vect}[1]{\overrightarrow{#1}}}
}{
    \CustomizeMathJax{\newcommand{\vect}[1]{\boldvectcommand{#1}}}
}
\ifdefstring{\probastyle}{\mathbb}{
    \CustomizeMathJax{\newcommand{\probastyle}[1]{\mathbb{#1}}}
}{
    \CustomizeMathJax{\newcommand{\probastyle}[1]{\mathrm{#1}}}
}
\ifdefstring{\mathset}{\mathbb}{
    \CustomizeMathJax{\newcommand{\mathset}[1]{\mathbb{#1}}}
}{
    \CustomizeMathJax{\newcommand{\mathset}[1]{\mathbf{#1}}}
}
}
\end{warpMathJax}
```

File 309 **lwarp-mleftright.sty**

§ 418 Package mleftright

(*Emulates or patches code by* Heiko Oberdiek.)

`mleftright` (*Pkg*) mleftright is used as-is, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{mleftright}[2019/12/03]
```

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mleft}{\left}}
\CustomizeMathJax{\newcommand{\mright}{\right}}
\CustomizeMathJax{\newcommand{\mleftright}{}}
\CustomizeMathJax{\newcommand{\mleftrightrestore}{}}
\end{warpMathJax}
```

File 310 **lwarp-morefloats.sty**

§ 419 Package morefloats

morefloats (*Pkg*) morefloats is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{morefloats}[2015/07/22]
```

File 311 **lwarp-moreverb.sty**

§ 420 Package moreverb

(*Emulates or patches code by* Robin Fairbairns.)

moreverb (*Pkg*) moreverb is supported with some patches.

```
\LWR@ProvidesPackagePass{moreverb}[2008/06/03]

\BeforeBeginEnvironment{verbatimtab}{%
\LWR@forcenewpage
\LWR@atbeginverbatim{Verbatim}%
}
\AfterEndEnvironment{verbatimtab}{%
\LWR@afterendverbatim%
}


\LetLtxMacro\LWRMV@orig@verbatimtabinput\@verbatimtabinput

\renewcommand{\@verbatimtabinput}[2][]{%
\LWR@forcenewpage
\LWR@atbeginverbatim{Verbatim}%
\LWRMV@orig@verbatimtabinput[#1]{#2}%
\LWR@afterendverbatim%
}

\BeforeBeginEnvironment{listing}{%
\LWR@forcenewpage
\LWR@atbeginverbatim{programlisting}%
}

\AfterEndEnvironment{listing}{%
\LWR@afterendverbatim%
}

\BeforeBeginEnvironment{listingcont}{%
\LWR@forcenewpage
```

```
\LWR@atbeginverbatim{programlisting}%
}

\AfterEndEnvironment{listingcont}{%
\LWR@afterendverbatim%
}

\LetLtxMacro\LWRMV@@listinginput\@listinginput

\renewcommand{\@listinginput}[3][]{
\LWR@forcenewpage
\LWR@atbeginverbatim{programlisting}%
\LWRMV@@listinginput[#1]{#2}{#3}%
\LWR@afterendverbatim%
}


\renewenvironment*{boxedverbatim}
{
\LWR@forcenewpage
\LWR@atbeginverbatim{boxedverbatim}%
\verbatim%
}
{
\endverbatim%
\LWR@afterendverbatim%
}
```

File 312 **lwarp-movie15.sty**

§ 421 Package movie15

movie15 (*Pkg*) movie15 is emualted.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 `\addmediapath` is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each `<audio>` and `<video>` object.

media9 slideshows are not supported.

`\hyperlinkmovie`, `\movieref`, and `\mediabutton` are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with …`/embed/`… instead of …`/v/`…

for HTML output:

```
\LWR@ProvidesPackageDrop{movie15}[2012/05/16]

\LWR@origRequirePackage{lwarp-common-multimedia}

\RequirePackage{xkeyval}

\newcommand*{\LWR@moviefifteen@text}{}

\define@key{LWR@moviefifteen}{text}{\renewcommand{\LWR@moviefifteen@text}{#1}}

\newcommand*{\LWR@includemovieb}[4][]{%
    \renewcommand{\LWR@moviefifteen@text}{(multimedia)}
    \setkeys*{LWR@moviefifteen}{#1}%
    \LWR@multimediab[#1,width=#2,height=#3]{\LWR@moviefifteen@text}{#4}%
}

\newrobustcmd*{\includemovie}{%
    \begingroup%
    \LWR@linkmediacatcodes%
    \LWR@includemovieb%
}


\newcommand*{\movieref}[3][]{}

\LetLtxMacro\movie\LWR@multimedia
% \LetLtxMacro\sound\LWR@multimedia% not in media15

\newcommand{\hyperlinkmovie}[3][]{}
```

File 313 **lwarp-mparhack.sty**

§ 422 Package mparhack

mparhack (*Pkg*) mparhack is ignored.

for HTML output: Discard all options for lwarp-mparhack:

```
\LWR@ProvidesPackageDrop{mparhack}[2005/04/17]
```

File 314 **lwarp-multibib.sty**

§ 423 Package multibib

(*Emulates or patches code by* Thorsten Hansen.)

multibib (*Pkg*) multibib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{multibib}[2008/12/10]

\xpatchcmd{\newcites}
    {{\@suffix}}
    {{\@suffix_html}}
    {}
    {\LWR@patcherror{multibib}{newcites}}
```

File 315 **lwarp-multicap.sty**

§ 424 Package multicap

multicap (*Pkg*) multicap is emualted.

for HTML output:

```
\LWR@ProvidesPackageDrop{multicap}[2002/05/04]

\newcommand*{\mfcaption}{\captionof{figure}}
\newcommand*{\mtcaption}{\captionof{table}}
\newcounter{mcapsize}
\newcounter{mcapskip}
\newlength{\abvmcapskip}
\newlength{\blwmcapskip}
```

File 316 **lwarp-multicol.sty**

§ 425 Package multicol

(*Emulates or patches code by* Frank Mittelbach.)

multicol (*Pkg*) multicol is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{multicol}[2021/10/28]
```

Multicols are converted into a 1–3 column display, browser-supported.

The optional multicols heading is placed inside a of class multicolsheading.

The content is placed inside a of class multicols.

Env multicols * {⟨*numcols*⟩} [⟨*heading*⟩]

```
\NewDocumentEnvironment{multicols}{s m o}
```

HTML class to contain everything:

```
{
    \LWR@forcenewpage
    \BlockClass{multicols}
```

Optional HTML class for the heading:

```
    \IfValueT{#3}{\begin{BlockClass}{multicolsheading}#3\end{BlockClass}}%
```

Change \linewidth to compensate for expected size:

```
    \setlength{\linewidth}{\linewidth/#2}
```

Locally force any minipages to be fullwidth:

```
    \booltrue{LWR@forceminipagefullwidth}
}
```

When done with the environment, close the :

```
{\endBlockClass}
```

Emulated null functions which are not used in HTML:

```
\newcommand*{\columnbreak}{}
\newcommand*{\newcolumn}{}
\newcommand*{\RLmulticolcolumns}{}
\newcommand*{\LRmulticolcolumns}{}

\newlength{\premulticols}
\newlength{\postmulticols}
\newlength{\multicolsep}
\newlength{\multicolbaselineskip}
\newlength{\multicoltolerance}
\newlength{\multicolpretolerance}
\newcommand*{\columnseprulecolor}{\normalcolor}
\newcounter{columnbadness}
\newcounter{finalcolumnbadness}
\newcounter{collectmore}
\newcounter{unbalance}
\newlength{\multicolovershoot}
\newlength{\multicolundershoot}


\NewDocumentCommand{\docolaction}{s o m m m}{%
    \IfValueTF{#2}{#2}{#3}%
}
```

File 317 **lwarp-multicolrule.sty**

§ 426 Package

multicolrule

multicolrule (*Pkg*) multicolrule is ignored.

for HTML output:

```
\RequirePackage{multicol}

\LWR@ProvidesPackageDrop{multicolrule}[2019/01/01]


\newcommand*{\SetMCRule}[1]{}
\NewDocumentCommand{\DeclareMCRulePattern}{m m}{}
```

File 318 **lwarp-multimedia.sty**

§ 427 Package

multimedia

`multimedia` (*Pkg*) multimedia is emulated.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for `.mp3` and `.mp4` files.

HTML5 <embed> objects are created for `http` and `ftp` links.

`\href` links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each `addresource=`, as well as each `flashvars` `source=` and `src=`. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside `\warpprintonly` or the `warpprint` environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the `text` option is supported to specify the poster text.

The `width`, `height`, and `totalheight` options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 `\addmediapath` is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

`\hyperlinkmovie`, `\movieref`, and `\mediabutton` are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with `.../embed/...` instead of `.../v/...`

for HTML output:

```
\LWR@ProvidesPackageDrop{multimedia}[2012/05/02]

\LWR@origRequirePackage{lwarp-common-multimedia}

\LetLtxMacro\movie\LWR@multimedia
\LetLtxMacro\sound\LWR@multimedia

\newcommand{\hyperlinkmovie}[3][]{}

\newcommand{\hyperlinksound}[3][]{}

```

```
\newcommand{\hyperlinkmute}
```

File 319 **lwarp-multiobjective.sty**

§ 428 Package multiobjective

(*Emulates or patches code by* Luis Martí.)

multiobjective (*Pkg*) multiobjective is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{multiobjective}[2008/08/19]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\dom}{\prec}}
\CustomizeMathJax{\newcommand{\negdom}{\not\prec}}
\CustomizeMathJax{\newcommand{\weakdom}{\preccurlyeq}}
\CustomizeMathJax{\newcommand{\negweakdom}{\not\preccurlyeq}}

\CustomizeMathJax{\newcommand{\strictdom}{\mathord{\prec}\!\!\!\mathord{\prec}}}
\CustomizeMathJax{\newcommand{\negstrictdom}{\mathord{\not\prec}\!\!\!\mathord{\prec}}}
\CustomizeMathJax{\newcommand{\multepsilondom}{\preccurlyeq_{\epsilon\cdot}}}
\CustomizeMathJax{\newcommand{\addiepsilondom}{\preccurlyeq_{\epsilon +}}}
\CustomizeMathJax{\newcommand{\better}{\triangleleft}}
\CustomizeMathJax{\def\vec#1{%
    \mathchoice%
        {{\displaystyle\boldsymbol{#1}}%
        {{\textstyle\boldsymbol{#1}}%
        {{\scriptstyle\boldsymbol{#1}}%
        {{\scriptscriptstyle\boldsymbol{#1}}%
}}

\CustomizeMathJax{\newcommand{\set}[1]{%
    \mathchoice%
        {{\displaystyle\mathcal{#1}}%
        {{\textstyle\mathcal{#1}}%
        {{\scriptstyle\mathcal{#1}}%
        {{\scriptscriptstyle\mathcal{#1}}%
}}
\CustomizeMathJax{\def\argmax{\mathop{{\mathrm{arg}}\,\max}}}
\CustomizeMathJax{\def\argmin{\mathop{{\mathrm{arg}}\,\min}
}}
\end{warpMathJax}
```

File 320 **lwarp-multirow.sty**

§ 429 Package multirow

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

multirow (*Pkg*) multirow is emulated during HTML output, and used as-is while inside a lateximage.

vposn

- Note that recent versions of multirow include a new optional vposn argument.

multirow cells

- For multirow, insert `\mrowcell` into any empty multi-row cells. This will be a null function for the print output, and is a placeholder for parsing the table for HTML output. An error is generated if this is missed.

```
... &  \multirow{2}{.5in}{text}  & ...
... &  \mrowcell                 & ...
```

colored cells

- The multirow documentation regarding colored cells recommends using a negative number of rows. This will not work with lwarp, so `\warpprintonly` and `\warpHTMLonly` must be used to make versions for print and HTML.

with `\multicolumn`

- See section 429.2 for `\multicolumrow`.

⚠ `\multicolumn` & `\multirow`

lwarp does not support directly combining `\multicolumn` and `\multirow`. Use `\multicolumnrow` instead. To create a 2 column, 3 row cell:

```
\multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}
```

The two arguments for `\multicolumn` come first, followed by the five arguments for `\multirow`, many of which are optional, followed by the contents.

⚠ skipped cells

⚠ empty cells

As per `\multirow`, skipped cells to the right of the `\multicolumnrow` statement are not included in the source code on the same line. On the following lines, `\mcolrowcell` must be used for each cell of each column and each row to be skipped. An error is generated if this is missed.

```
... &  \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}  & ...
... &  \mcolrowcell          &          \mcolrowcell   & ...
... &  \mcolrowcell          &          \mcolrowcell   & ...
```

⚠ MathJax

- MathJax does not support multirow, so it is emulated to only print its text on the first row. `\multirow` works as expected in text `tabulars` or SVG math.

In a `latexmage`, the print versions are restored.

See section 75.24 for the print-mode versions.

for HTML output:

Remove the placeholder macro which was used if multirow was not loaded:

```
\LetLtxMacro\multirow\relax
\LWR@ProvidesPackagePass{multirow}[2021/03/15]
```

`\LWR@multirowborder`

Set to `left` or `right` to create a thick border for the cell, for use by bigdelim:

```
\newcommand{\LWR@multirowborder}{}
```

§ 429.1 Multirow

\LWR@multirow@par

\par inside a \multirow.

```
\newcommand*{\LWR@multirow@par}{%
    \LWR@htmltag{br /}%
}%
```

\multirow [⟨*1: vpos*⟩] {⟨*2: numrows*⟩} [⟨*3:bigstruts*⟩] {⟨*4: width*⟩} [⟨*5: vmove*⟩] {⟨*6: text*⟩}

```
\NewDocumentCommand{\LWR@HTML@multirow}{O{c} m o m o +m}%
{%
    \LWR@traceinfo{LWR@HTML@multirow #1 #2 #4}%
    \booltrue{LWR@usedmultirow}%
    \LWR@maybenewtablerow%
    \LWR@tabularleftedge%
```

Print the start of a new table data cell:

```
    \LWR@htmltag{%
        td rowspan=\textquotedbl#2\textquotedbl\ %
```

A class adds the column spec and the rule:

```
        class=\textquotedbl{}td%
```

Append this column's spec:

```
       \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag. Also add the vertical bar class.

```
        \LWR@addcmidruletrim%
        \LWR@addleftmostbartag%
        \LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
        \textquotedbl%
        \LWR@tdstartstyles%
```

The vertical alignment, if given:

```
    \ifstrequal{#1}{c}{\LWR@tdaddstyle\LWR@print@mbox{vertical-align:middle}}{}%
    \ifstrequal{#1}{b}{\LWR@tdaddstyle\LWR@print@mbox{vertical-align:bottom}}{}%
    \ifstrequal{#1}{t}{\LWR@tdaddstyle\LWR@print@mbox{vertical-align:top}}{}%
```

The left/right border, if given:

```
        \ifdefvoid{\LWR@multirowborder}{}{%
            \LWR@tdaddstyle%
           \LWR@print@mbox{border-\LWR@multirowborder:} 2px dotted black ; %
            \LWR@print@mbox{padding-\LWR@multirowborder:} 2px%
        }%
```

Additional style elements:

```
        \LWR@addcmidrulewidth%
        \LWR@addcdashline%
        \LWR@addtabularrulecolors%
        \LWR@tdendstyles%
    }%
```

The column's < spec:

```
    \LWR@getexparray{LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
```

While printing the text, redefine \\ to generate a new line. If a nested tabular occurs, \\ is redefined to \LWR@tabularendofline at the start of the tabular, then \LWR@endofline before again printing any \multirow contents inside the nested tabular.

\par is redefined to insert an HTML break, and if tabular is nested, it is redefined at the start of tabular.

```
    \begingroup%
        \LetLtxMacro{\\}{\LWR@endofline}%
        \booltrue{LWR@in@multirow@par}%
        #6%
    \endgroup%
    \LWR@stoppars%
    \boolfalse{LWR@intabularmetadata}%
    \renewcommand{\LWR@multirowborder}{}%
    \LWR@traceinfo{LWR@HTML@multirow done}%
}%

\LWR@formatted{multirow}
```

§ 429.2 Combined multicolumn and multirow

\multicolumnrow {⟨*1:cols*⟩} {⟨*2:halign*⟩} [⟨*3:vpos*⟩] {⟨*4:numrows*⟩} [⟨*5:bigstruts*⟩] {⟨*6:width*⟩} [⟨*7:fixup*⟩] {⟨*8:text*⟩}

\IfPackageLoadedTF{multirow} determines if v2.0 or later of multirow was used, which included the \ProvidesPackage macro.

The HTML version follows.

\AtBeginDocument because the print version had to see if multirow was loaded before determining how to define \LWR@print@multicolumnrow.

```
\AtBeginDocument{

\NewExpandableDocumentCommand{\LWR@HTML@multicolumnrow}{m m O{} m O{} m O{} +m}{%

\booltrue{LWR@usedmultirow}%
```

Figure out how many extra HTML columns to add for @ and ! columns:

```
\LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}
```

Create the multicolumn/multirow tag, temporarily redefining the end of line. (Using a group caused problems with a nested `tabular`.

```
\LetLtxMacro{\\}{\LWR@endofline}%
\LWR@domulticolumn[#3][#4]{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#8}%
\LetLtxMacro{\\}{\LWR@tabularendofline}%
```

Move to the next LaTeX column:

```
\defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
\defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing `@` or `!` columns for this cell:

```
\booltrue{LWR@skipatbang}%
}

\LWR@expandableformatted{multicolumnrow}

}% \AtBeginDocument
```

For MathJax. Only the text is used. All other parameters are ignored.

```
\begin{warpMathJax}
% \multirow[vpos]{num}[bigstruts]{width}[vmove]{text}
\CustomizeMathJax{\newcommand{\LWRsubmultirow}[2][]{#2}}
\CustomizeMathJax{\newcommand{\LWRmultirow}[2][]{\LWRsubmultirow}}
\CustomizeMathJax{\newcommand{\multirow}[2][]{\LWRmultirow}}
%
\CustomizeMathJax{\newcommand{\mrowcell}{}}
\CustomizeMathJax{\newcommand{\mcolrowcell}{}}
\CustomizeMathJax{\newcommand{\STneed}[1]{}}
\end{warpMathJax}
```

File 321 **lwarp-multitoc.sty**

§ 430 Package multitoc

`multitoc` (*Pkg*) multitoc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{multitoc}[1999/06/08]

\newcommand{\multicolumntoc}{2}
\newcommand{\multicolumnlot}{2}
\newcommand{\multicolumnlof}{2}
\newcommand*{\immediateaddtocontents}[2]{}
```

File 322 **lwarp-musicography.sty**

§ 431 Package musicography

(*Emulates or patches code by* Andrew A. Cashner.)

musicography (*Pkg*) musicography is patched for use by lwarp.

Images are used for the meter symbols and fingered bass, since the HTML fonts tend not to be the correct size and HTML cannot stack items. The HTML `alt` tag copies `C` and `3/2`, etc. Hashes are used for the meter images, which are then reused as necessary.

⚠ Note that browser support for musical symbols may be buggy. ALT text and copy/paste into a text editor work well.

for HTML output:

```
\LWR@ProvidesPackagePass{musicography}[2019/05/28]

\NewDocumentCommand{\LWR@HTML@musSymbol}{ O{\musFont} m m m m }{%
\begin{lateximage}%
{#1\kern#2\raisebox{#3}{#5}\kern#4}%
\end{lateximage}%
}

\LWR@formatted{musSymbol}

\NewDocumentCommand{\LWR@HTML@musStemmedNote}{ m }{%
\begin{lateximage}%
\musSymbol{0.05em}{0.5ex}{0.2em}{#1\musStem}%
\end{lateximage}%
}

\LWR@formatted{musStemmedNote}

\NewDocumentCommand{\LWR@HTML@musFlaggedNote}{ m m }{%
\begin{lateximage}%
\musSymbol{0.05em}{0.5ex}{0pt}{#1\musStem}%
\musSymbol{0pt}{0pt}{0.9em}{#2}%
\end{lateximage}%
}

\LWR@formatted{musFlaggedNote}

\NewDocumentCommand{\LWR@HTML@musDottedNote}{ m }{%
\begin{lateximage}%
#1\musDot%
\end{lateximage}%
}

\LWR@formatted{musDottedNote}

\NewDocumentCommand{\LWR@HTML@musMeter}{ m m }{%
\begin{lateximage}*[#1/#2][#1#2]*%
\musStack{#1 #2}\kern0.05em%
\end{lateximage}%
}

\LWR@formatted{musMeter}

\NewDocumentCommand{\LWR@HTML@meterCplus}{ m }{%
\begin{lateximage}*[C#1]*%
    \meterC{}\kern-0.7pt#1%
\end{lateximage}%
}

\LWR@formatted{meterCplus}
```

```
\NewDocumentCommand{\LWR@HTML@meterC}{}{%
\begin{lateximage}*[C]*%
\musSymbolMeter{\symbol{83}}%
\end{lateximage}%
}

\LWR@formatted{meterC}

\NewDocumentCommand{\LWR@HTML@meterCutC}{}{%
\begin{lateximage}*[C|]*%
\musSymbolMeter{\symbol{82}}%
\end{lateximage}%
}

\LWR@formatted{meterCutC}

\NewDocumentCommand{\LWR@HTML@meterCThreeTwo}{}{%
\begin{lateximage}*[C3/2]*%
\meterCplus{\musStack{3 2}}%
\end{lateximage}%
}

\LWR@formatted{meterCThreeTwo}

\NewDocumentCommand{\LWR@HTML@meterO}{}{\HTMLunicode{25EF}}

\LWR@formatted{meterO}

\newcommand{\LWR@null@noFig}[1][]{}%

\NewDocumentCommand{\LWR@HTML@musFig}{ m }{%
\begin{lateximage}*[%
    {% ALT text for copy/paste
        \LetLtxMacro\noFig\LWR@null@noFig%
        \LetLtxMacro\musSharp\LWR@HTML@musSharp%
        \LetLtxMacro\musDoubleSharp\LWR@HTML@musDoubleSharp%
        \LetLtxMacro\musFlat\LWR@HTML@musFlat%
        \LetLtxMacro\musDoubleFlat\LWR@HTML@musDoubleFlat%
        \LetLtxMacro\musNatural\LWR@HTML@musNatural%
        {#1}% braces here because \noFig uses []
    }%
]*%
    \musStack[\musFigFont]{#1}%
\end{lateximage}%
}

\LWR@formatted{musFig}

\NewDocumentCommand{\LWR@HTML@musFlat}       {}{\HTMLunicode{266D}}
\NewDocumentCommand{\LWR@HTML@musDoubleFlat} {}{\HTMLunicode{1D12B}}
\NewDocumentCommand{\LWR@HTML@musSharp}      {}{\HTMLunicode{266F}}
\NewDocumentCommand{\LWR@HTML@musDoubleSharp}{}{\HTMLunicode{1D12A}}
\NewDocumentCommand{\LWR@HTML@musNatural}    {}{\HTMLunicode{266E}}

\LWR@formatted{musFlat}
\LWR@formatted{musDoubleFlat}
\LWR@formatted{musSharp}
\LWR@formatted{musDoubleSharp}
\LWR@formatted{musNatural}
```

```
\NewDocumentCommand{\LWR@HTML@musWhole}         {}{\HTMLunicode{1D15D}}
\NewDocumentCommand{\LWR@HTML@musHalf}          {}{\HTMLunicode{1D15E}}
\NewDocumentCommand{\LWR@HTML@musQuarter}       {}{\HTMLunicode{1D15F}}
\NewDocumentCommand{\LWR@HTML@musEighth}        {}{\HTMLunicode{1D160}}
\NewDocumentCommand{\LWR@HTML@musSixteenth}     {}{\HTMLunicode{1D161}}
\NewDocumentCommand{\LWR@HTML@musThirtySecond}  {}{\HTMLunicode{1D162}}
\NewDocumentCommand{\LWR@HTML@musSixtyFourth}   {}{\HTMLunicode{1D163}}

\LWR@formatted{musWhole}
\LWR@formatted{musHalf}
\LWR@formatted{musQuarter}
\LWR@formatted{musEighth}
\LWR@formatted{musSixteenth}
\LWR@formatted{musThirtySecond}
\LWR@formatted{musSixtyFourth}

\NewDocumentCommand{\LWR@HTML@musWholeDotted}{}
    {\HTMLunicode{1D15D}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musHalfDotted}{}
    {\HTMLunicode{1D15E}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musQuarterDotted}{}
    {\HTMLunicode{1D15F}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musEighthDotted}{}
    {\HTMLunicode{1D160}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musSixteenthDotted}{}
    {\HTMLunicode{1D161}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musThirtySecondDotted}{}
    {\HTMLunicode{1D162}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musSixtyFourthDotted}{}
    {\HTMLunicode{1D163}\HTMLunicode{1D16D}}

\LWR@formatted{musWholeDotted}
\LWR@formatted{musHalfDotted}
\LWR@formatted{musQuarterDotted}
\LWR@formatted{musEighthDotted}
\LWR@formatted{musSixteenthDotted}
\LWR@formatted{musThirtySecondDotted}
\LWR@formatted{musSixtyFourthDotted}
```

File 323 **lwarp-mwe.sty**

§ 432 Package mwe

(*Emulates or patches code by* Martin Scharrer.)

mwe (*Pkg*) mwe is used as-is, but a warning is issued to copy the images to the local directory.

for HTML output:

```
\LWR@ProvidesPackagePass{mwe}[2018/03/30]

\AtEndDocument{%
    \PackageWarningNoLine{lwarp}{%
        For package mwe, copy any mwe images to be used for\MessageBreak
        HTML, such as PNG or JPG, to the document's base\MessageBreak
        directory.  Neither a subdirectory nor the mwe\MessageBreak
        directory will work, due to the TeX file search\MessageBreak
        algorithm%
```

```
    }%
}%
```

File 324 **lwarp-nameauth.sty**

§ 433 Package nameauth

(*Emulates or patches code by* Charles P. Schaum.)

nameauth (*Pkg*) nameauth is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{nameauth}[2017/03/22]
```

\@nameauth@Hook (*Hook*) [nameauth] lwarp formatting is inserted.

```
\renewcommand*\@nameauth@Hook[1]
{%
  \if@nameauth@Lock
    \@nameauth@InHooktrue%
    \protected@edef\test{#1}%
    \expandafter\@nameauth@TestDot\expandafter{\test}%
    \if@nameauth@InAKA
      \if@nameauth@AlwaysFormat
        \@nameauth@FirstFormattrue%
      \else
        \unless\if@nameauth@AKAFormat
        \@nameauth@FirstFormatfalse\fi
      \fi
      \if@nameauth@MainFormat
        \if@nameauth@FirstFormat
          \bgroup\NamesFormat{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \else
          \bgroup\MainNameHook{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \fi
      \else
        \if@nameauth@FirstFormat
          \bgroup\FrontNamesFormat{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \else
          \bgroup\FrontNameHook{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \fi
      \fi
    \else
      \if@nameauth@AlwaysFormat
        \@nameauth@FirstFormattrue%
      \fi
      \if@nameauth@MainFormat
        \if@nameauth@FirstFormat
          \bgroup\NamesFormat{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
```

```
          }\egroup%
        \else
          \bgroup\MainNameHook{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \fi
      \else
        \if@nameauth@FirstFormat
          \bgroup\FrontNamesFormat{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \else
          \bgroup\FrontNameHook{%
            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
          }\egroup%
        \fi
      \fi
    \fi
    \@nameauth@FirstFormatfalse%
    \@nameauth@InHookfalse%
  \fi
}
```

File 325 **lwarp-nameref.sty**

§ 434 Package nameref

nameref (*Pkg*) nameref is emulated by lwarp.

for HTML output: Discard all options for lwarp-nameref:

```
\PackageInfo{lwarp}{%
Using the lwarp HTML version of package ‘nameref’,\MessageBreak
and discarding options.\MessageBreak
(Not using \protect\ProvidesPackage, so that other packages\MessageBreak
do not attempt to patch lwarp’s version of ‘nameref’.)\MessageBreak
}
\DeclareOption*{}
\ProcessOptions\relax
```

File 326 **lwarp-natbib.sty**

§ 435 Package natbib

(*Emulates or patches code by* Patrick W. Daly.)

natbib (*Pkg*) natbib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{natbib}[2010/09/13]
```

Replace math < and > with \textless and \textgreater:

A macro to compare:

```
\newcommand{\LWRNB@NAT@open}{$<$}
```

To patch \NAT@open and \NAT@close

```
\newcommand{\LWRNB@patchnatbibopenclose}{
\ifdefstrequal{\NAT@open}{\LWRNB@NAT@open}
{
    \renewcommand{\NAT@open}{\textless}
    \renewcommand{\NAT@close}{\textgreater}
}{}
}
```

Do it now in case angle was selected as an option:

```
\LWRNB@patchnatbibopenclose
```

Also patch \setcitestyle to patch after settings are made:

```
\let\LWRNB@origsetcitestyle\setcitestyle

\renewcommand{\setcitestyle}[1]{%
\LWRNB@origsetcitestyle{#1}%
\LWRNB@patchnatbibopenclose%
}
```

Syncronize the autopage labels:

```
\xpretocmd{\NAT@reset@parser}
    {\LWR@newautopagelabel{page}}%
    {}
    {\LWR@patcherror{natbib}{NAT@reset@parser}}
```

File 327 **lwarp-nccfancyhdr.sty**

§ 436 Package

nccfancyhdr

(*Emulates or patches code by* Alexander I. Rozhenko.)

nccfancyhdr (*Pkg*) nccfancyhdr is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nccfancyhdr}[2004/12/07]

\newcommand*{\headrulewidth}{}
\newcommand*{\footrulewidth}{}
\newcommand{\headstrutheight}{}
\newcommand{\footstrutheight}{}
\newcommand*{\headrule}{}
\newcommand*{\footrule}{}

\newdimen\headwidth
\newcommand*{\extendedheaders}{}
\newcommand*{\normalheaders}{}

\newcommand*{\fancyhead}[2][]{}
\newcommand*{\fancyfoot}[2][]{}
\newcommand*{\fancyhf}[2][]{}
\newcommand*{\fancypagestyle}[2]{}
\newcommand*{\lhead}[2][]{}
```

```
\newcommand*{\chead}[2][]{}
\newcommand*{\rhead}[2][]{}
\newcommand*{\lfoot}[2][]{}
\newcommand*{\cfoot}[2][]{}
\newcommand*{\rfoot}[2][]{}

\newcommand{\nouppercase}[1]{#1}

\NewDocumentCommand{\fancycenter}{o o m m m}{}

\NewDocumentCommand{\newpagestyle}{m o m}{}

\newcommand*{\iffloatpage}[2]{#2}
\newcommand*{\ifftopfloat}[2]{#2}
\newcommand*{\iffbotfloat}[2]{#2}
```

File 328 **lwarp-nccfoots.sty**

§ 437 Package nccfoots

(*Emulates or patches code by* Alexander I. Rozhenko.)

nccfoots (*Pkg*) nccfoots is used as-is, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{nccfoots}[2005/02/03]
```

To nullify the footnotes where necessary:

```
\apptocmd{\LWR@nullifyfootnotes}{%
    \renewcommand*{\Footnote}[1]{}%
    \renewcommand*{\Footnotemark}[1]{}%
}{}{}
```

⚠ For MathJax. There is no way to test for an empty argument, so the mark is not automatically duplicated.

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\Footnotemark}[1]{{}^{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\Footnote}[2]{\Footnotemark{#1}}}
\end{warpMathJax}
```

File 329 **lwarp-nccmath.sty**

§ 438 Package nccmath

(*Emulates or patches code by* Alexander I. Rozhenko.)

nccmath (*Pkg*) nccmath is patched for use by lwarp, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{nccmath}[2006/01/20]

\let\LWR@origeqnarray\eqnarray
\let\LWR@origendeqnarray\endeqnarray

```

```
\csletcs{LWR@origeqnarraystar}{eqnarray*}
\csletcs{LWR@origendeqnarraystar}{endeqnarray*}

\RenewEnviron{eqnarray}
{%

    \LWR@eqnarrayfactor

}

\RenewEnviron{eqnarray*}
{%

    \begingroup
    \csletcs{LWR@origeqnarray}{LWR@origeqnarraystar}
    \csletcs{LWR@origendeqnarray}{LWR@origendeqnarraystar}
    \boolfalse{LWR@numbereqnarray}
    \LWR@eqnarrayfactor
    \endgroup

}

\def\eqs{%
    \@ifstar\LWR@nccmath@eqsstar\LWR@nccmath@eqs%
}
\newcommand*{\LWR@nccmath@eqsstar}[2][]{\begin{eqnarray*}#2\end{eqnarray*}}
\newcommand*{\LWR@nccmath@eqs}[2][]{\begin{eqnarray}#2\end{eqnarray}}

\begin{warpMathJax}

\CustomizeMathJax{\renewcommand{\intertext}[2][]{\text{#2}\notag \\}}
\CustomizeMathJax{\newenvironment{fleqn}[1][]{}{}}
\CustomizeMathJax{\newenvironment{ceqn}{}{}}
\CustomizeMathJax{\newenvironment{darray}[2][c]{\begin{array}[#1]{#2}}{\end{array}}}
\CustomizeMathJax{\newcommand{\dmulticolumn}[3]{#3}}
```

As of v0.86, MathJax v3 does not offer `\\*`, so the unstarred version is used here.

```
\CustomizeMathJax{\newcommand{\LWRnrnostar}[1][0.5ex]{\\[#1]}}
\CustomizeMathJax{\newcommand{\nr}{\ifstar\LWRnrnostar\LWRnrnostar}}

\CustomizeMathJax{\newcommand{\mrel}[1]{\begin{aligned}#1\end{aligned}}}
\CustomizeMathJax{\newcommand{\underrel}[2]{\underset{#2}{#1}}}
\CustomizeMathJax{\newcommand{\medmath}[1]{#1}}
\CustomizeMathJax{\newcommand{\medop}[1]{#1}}
\CustomizeMathJax{\newcommand{\medint}[1]{#1}}
\CustomizeMathJax{\newcommand{\medintcorr}[1]{#1}}
\CustomizeMathJax{\newcommand{\mfrac}[2]{\frac{#1}{#2}}}
\CustomizeMathJax{\newcommand{\mbinom}[2]{\binom{#1}{#2}}}
\CustomizeMathJax{\newenvironment{mmatrix}{\begin{matrix}}{\end{matrix}}}

\CustomizeMathJax{\newcommand{\displaybreak}[1][]{}}
```

`\eq`, `\eqs`, `\eqalign` are created by LaTeX, not MathJax.

```
\end{warpMathJax}
```

File 330 **lwarp-needspace.sty**

§ 439 Package needspace

(*Emulates or patches code by* Peter Wilson.)

needspace (*Pkg*) needspace is ignored.

for HTML output: Discard all options for lwarp-needspace:

```
\LWR@ProvidesPackageDrop{needspace}[2010/09/12]

\DeclareDocumentCommand{\needspace}{m}{}
\DeclareDocumentCommand{\Needspace}{s m}{}
```

File 331 **lwarp-newpxmath.sty**

§ 440 Package newpxmath

(*Emulates or patches code by* Michael Sharpe.)

newpxmath (*Pkg*) newpxmath is used as-is for svg math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options, except `slantedGreek` is honored. The dedicated macros for upright and italic Greek do work correctly.

svg math should appear the same as the printed output.

for HTML output: The MathJax code from newtxmath is used:

```
\LWR@ProvidesPackagePass{newpxmath}[2020/01/09]

\LWR@infoprocessingmathjax{newpxmath}

\LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

%   * \marg{2: prefix} \marg{3: postfix} \marg{4: i/u: italic/upright}
\LWR@mathjax@addgreek@u@up*{}{up}
\LWR@mathjax@addgreek@u@up*{up}{}
\LWR@mathjax@addgreek@l@up{up}{}
\LWR@mathjax@addgreek@l@up{}{up}
\LWR@mathjax@addgreek@u@it*{}{it}
\LWR@mathjax@addgreek@l@it{}{it}
```

Optional slanted Greek:

```
\ifpx@slantedG
    \LWR@mathjax@addgreek@u@it*{}{}
\fi

```

```
\end{warpMathJax}
```

File 332 **lwarp-newtxmath.sty**

§ 441 Package newtxmath

(*Emulates or patches code by* Michael Sharpe.)

newtxmath (*Pkg*) newtxmath is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options, except slantedGreek is honored, and except that bold italic Latin letters are not defined for MathJax if the option is not selected.

The dedicated macros for upright and italic Greek and bold italic Latin letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{newtxmath}[2020/08/04]

\LWR@infoprocessingmathjax{newtxmath}

\LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

%   * \marg{2: prefix} \marg{3: postfix} \marg{4: i/u: italic/upright}
\LWR@mathjax@addgreek@u@up*{}{up}
\LWR@mathjax@addgreek@u@up*{up}{}
\LWR@mathjax@addgreek@l@up{up}{}
\LWR@mathjax@addgreek@l@up{}{up}
\LWR@mathjax@addgreek@u@it*{}{it}
\LWR@mathjax@addgreek@l@it{}{it}

% only newtxmath, not newpxmath:
\LWR@mathjax@addgreek@u@it*{it}{}
\LWR@mathjax@addgreek@l@it{it}{}

% only newtxmath, not newpxmath:
\ifdef{\iftx@BI}{
    \iftx@BI
        \LWR@mathjax@addlatin@u@bfit{BI}
        \LWR@mathjax@addlatin@l@bfit{BI}
    \fi
}{}
```

Optional slanted Greek:

```
\iftx@slantedG
    \LWR@mathjax@addgreek@u@it*{}{}
\fi

\end{warpMathJax}
```

File 333 **lwarp-newtxsf.sty**

§ 442 Package

newtxsf

(*Emulates or patches code by* Michael Sharpe.)

newtxsf (*Pkg*) newtxsf is used as-is for SVG math, and is emulated for MathJax.

⚠ limitations The MathJax emulation ignores all package options, except slantedGreek is honored. The dedicated macros for upright and italic Greek and bold italic Latin letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
\LWR@ProvidesPackagePass{newtxsf}[2020/05/02]

\LWR@infoprocessingmathjax{newtxsf}

\LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}

%   * \marg{2: prefix} \marg{3: postfix} \marg{4: i/u: italic/upright}
\LWR@mathjax@addgreek@u@up*{}{up}
\LWR@mathjax@addgreek@u@up*{up}{}
\LWR@mathjax@addgreek@l@up{up}{}
\LWR@mathjax@addgreek@l@up{}{up}
\LWR@mathjax@addgreek@u@it*{}{it}
\LWR@mathjax@addgreek@l@it{}{it}

% only newtxmath, not newpxmath:
\LWR@mathjax@addgreek@u@it*{it}{}
\LWR@mathjax@addgreek@l@it{it}{}
%
% only newtxmath, not newpxmath:
\ifdef{\iftx@BI}{
    \iftx@BI
        \LWR@mathjax@addlatin@u@bfit{BI}
        \LWR@mathjax@addlatin@l@bfit{BI}
    \fi
}{}
```

Optional slanted Greek:

```
\iftx@slantedG
    \LWR@mathjax@addgreek@u@it*{}{}
\fi

\end{warpMathJax}
```

File 334 **lwarp-nextpage.sty**

§ 443 Package nextpage

(*Emulates or patches code by* Peter Wilson.)

nextpage (*Pkg*) nextpage is ignored.

for HTML output: Discard all options for lwarp-nextpage.

```
\LWR@ProvidesPackageDrop{nextpage}[2009/09/03]

\DeclareDocumentCommand{\cleartoevenpage}{o}{}
\DeclareDocumentCommand{\movetoevenpage}{o}{}
\DeclareDocumentCommand{\cleartooddpage}{o}{}
\DeclareDocumentCommand{\movetooddpage}{o}{}
```

File 335 **lwarp-nfssext-cfr.sty**

§ 444 Package nfssext-cfr

(*Emulates or patches code by* Clea F. Rees.)

nfssext-cfr (*Pkg*) nfssext-cfr is emulated in HTML, and used as-is in print output.

Results depend on the browser's font.

for HTML output:

```
\LWR@ProvidesPackagePass{nfssext-cfr}[2017/03/28]
```

Macros which are present in the lwarp core are commented out here.

```
\newrobustcmd{\LWR@HTML@lnstyle}{}
\newrobustcmd{\LWR@HTML@osstyle}{\LWR@HTML@scshape}
\newrobustcmd{\LWR@HTML@instyle}{}
\newrobustcmd{\LWR@HTML@sustyle}{}
\newrobustcmd{\LWR@HTML@swstyle}{}
\newrobustcmd{\LWR@HTML@pstyle}{}
\newrobustcmd{\LWR@HTML@tistyle}{}
\newrobustcmd{\LWR@HTML@ostyle}{\LWR@HTML@scshape}
\newrobustcmd{\LWR@HTML@postyle}{\LWR@HTML@scshape}
\newrobustcmd{\LWR@HTML@ltstyle}{}
\newrobustcmd{\LWR@HTML@ofstyle}{}
\newrobustcmd{\LWR@HTML@altstyle}{}
\newrobustcmd{\LWR@HTML@regstyle}{}
\newrobustcmd{\LWR@HTML@embossstyle}{}
\newrobustcmd{\LWR@HTML@ornamentalstyle}{}
\newrobustcmd{\LWR@HTML@qtstyle}{}
\newrobustcmd{\LWR@HTML@shstyle}{}
\newrobustcmd{\LWR@HTML@swashstyle}{}
\newrobustcmd{\LWR@HTML@tmstyle}{\renewcommand*{\LWR@f@family}{tt}}
\newrobustcmd{\LWR@HTML@tvstyle}{\renewcommand*{\LWR@f@family}{tt}}
\newrobustcmd{\LWR@HTML@tstyle}{}
\newrobustcmd{\LWR@HTML@lstyle}{}
```

```
\newrobustcmd{\LWR@HTML@tlstyle}{}
\newrobustcmd{\LWR@HTML@plstyle}{}
\newrobustcmd{\LWR@HTML@tostyle}{\LWR@HTML@scshape}
% \newrobustcmd{\LWR@HTML@sishape}{}
\newrobustcmd{\LWR@HTML@olshape}{}
\newrobustcmd{\LWR@HTML@scolshape}{}
\newrobustcmd{\LWR@HTML@ushape}{}
\newrobustcmd{\LWR@HTML@scushape}{}
\newrobustcmd{\LWR@HTML@uishape}{\LWR@HTML@itshape}
\newrobustcmd{\LWR@HTML@rishape}{}
\newrobustcmd{\LWR@HTML@regwidth}{}
\newrobustcmd{\LWR@HTML@nwwidth}{}
\newrobustcmd{\LWR@HTML@cdwidth}{}
\newrobustcmd{\LWR@HTML@ecwidth}{}
\newrobustcmd{\LWR@HTML@ucwidth}{}
\newrobustcmd{\LWR@HTML@etwidth}{}
\newrobustcmd{\LWR@HTML@epwidth}{}
\newrobustcmd{\LWR@HTML@exwidth}{}
\newrobustcmd{\LWR@HTML@uxwidth}{}
\newrobustcmd{\LWR@HTML@mbweight}{\renewcommand*{\LWR@f@series}{md}}
\newrobustcmd{\LWR@HTML@dbweight}{\renewcommand*{\LWR@f@series}{db}}
\newrobustcmd{\LWR@HTML@sbweight}{\renewcommand*{\LWR@f@series}{sb}}
% \newrobustcmd{\LWR@HTML@ebweight}{\renewcommand*{\LWR@f@series}{eb}}
\newrobustcmd{\LWR@HTML@ubweight}{\renewcommand*{\LWR@f@series}{ub}}
% \newrobustcmd{\LWR@HTML@lgweight}{\renewcommand*{\LWR@f@series}{lg}}
\newrobustcmd{\LWR@HTML@elweight}{\renewcommand*{\LWR@f@series}{el}}
\newrobustcmd{\LWR@HTML@ulweight}{\renewcommand*{\LWR@f@series}{ul}}
% \newrobustcmd{\LWR@HTML@itshape}{}
% \newrobustcmd{\LWR@HTML@scshape}{}
% \newrobustcmd{\LWR@HTML@upshape}{}
\newrobustcmd{\LWR@HTML@dfshape}{}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \newrobustcmd{\LWR@HTML@swshape}{}
}

\newrobustcmd{\LWR@HTML@ornament}[1]{}

\LWR@formatted{lnstyle}
\LWR@formatted{osstyle}
\LWR@formatted{instyle}
\LWR@formatted{sustyle}
\LWR@formatted{swstyle}
\LWR@formatted{pstyle}
\LWR@formatted{tistyle}
\LWR@formatted{ostyle}
\LWR@formatted{postyle}
\LWR@formatted{ltstyle}
\LWR@formatted{ofstyle}
\LWR@formatted{altstyle}
\LWR@formatted{regstyle}
\LWR@formatted{embossstyle}
\LWR@formatted{ornamentalstyle}
\LWR@formatted{qtstyle}
\LWR@formatted{shstyle}
\LWR@formatted{swashstyle}
\LWR@formatted{tmstyle}
\LWR@formatted{tvstyle}
\LWR@formatted{tstyle}
\LWR@formatted{lstyle}
```

```
\LWR@formatted{tlstyle}
\LWR@formatted{plstyle}
\LWR@formatted{tostyle}
% \LWR@formatted{sishape}
\LWR@formatted{olshape}
\LWR@formatted{scolshape}
\LWR@formatted{ushape}
\LWR@formatted{scushape}
\LWR@formatted{uishape}
\LWR@formatted{rishape}
\LWR@formatted{regwidth}
\LWR@formatted{nwwidth}
\LWR@formatted{cdwidth}
\LWR@formatted{ecwidth}
\LWR@formatted{ucwidth}
\LWR@formatted{etwidth}
\LWR@formatted{epwidth}
\LWR@formatted{exwidth}
\LWR@formatted{uxwidth}
\LWR@formatted{mbweight}
\LWR@formatted{dbweight}
\LWR@formatted{sbweight}
% \LWR@formatted{ebweight}
\LWR@formatted{ubweight}
% \LWR@formatted{lgweight}
\LWR@formatted{elweight}
\LWR@formatted{ulweight}
\LWR@formatted{itshape}% adapt to the new print version
\LWR@formatted{scshape}% adapt to the new print version
\LWR@formatted{upshape}% adapt to the new print version
\LWR@formatted{dfshape}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \LWR@formatted{swshape}
}

\LWR@formatted{ornament}


\FilenameNullify{%
    \LetLtxMacro\lnstyle\@empty%
    \LetLtxMacro\osstyle\@empty%
    \LetLtxMacro\instyle\@empty%
    \LetLtxMacro\sustyle\@empty%
    \LetLtxMacro\swstyle\@empty%
    \LetLtxMacro\pstyle\@empty%
    \LetLtxMacro\tistyle\@empty%
    \LetLtxMacro\ostyle\@empty%
    \LetLtxMacro\postyle\@empty%
    \LetLtxMacro\ltstyle\@empty%
    \LetLtxMacro\ofstyle\@empty%
    \LetLtxMacro\altstyle\@empty%
    \LetLtxMacro\regstyle\@empty%
    \LetLtxMacro\embossstyle\@empty%
    \LetLtxMacro\ornamentalstyle\@empty%
    \LetLtxMacro\qtstyle\@empty%
    \LetLtxMacro\shstyle\@empty%
    \LetLtxMacro\swashstyle\@empty%
    \LetLtxMacro\tmstyle\@empty%
    \LetLtxMacro\tvstyle\@empty%
    \LetLtxMacro\tstyle\@empty%
```

```
    \LetLtxMacro\lstyle\@empty%
    \LetLtxMacro\tlstyle\@empty%
    \LetLtxMacro\plstyle\@empty%
    \LetLtxMacro\tostyle\@empty%
%   \LetLtxMacro\sishape\@empty%
    \LetLtxMacro\olshape\@empty%
    \LetLtxMacro\scolshape\@empty%
    \LetLtxMacro\ushape\@empty%
    \LetLtxMacro\scushape\@empty%
    \LetLtxMacro\uishape\@empty%
    \LetLtxMacro\rishape\@empty%
    \LetLtxMacro\regwidth\@empty%
    \LetLtxMacro\nwwidth\@empty%
    \LetLtxMacro\cdwidth\@empty%
    \LetLtxMacro\ecwidth\@empty%
    \LetLtxMacro\ucwidth\@empty%
    \LetLtxMacro\etwidth\@empty%
    \LetLtxMacro\epwidth\@empty%
    \LetLtxMacro\exwidth\@empty%
    \LetLtxMacro\uxwidth\@empty%
    \LetLtxMacro\mbweight\@empty%
    \LetLtxMacro\dbweight\@empty%
    \LetLtxMacro\sbweight\@empty%
%   \LetLtxMacro\ebweight\@empty%
    \LetLtxMacro\ubweight\@empty%
%   \LetLtxMacro\lgweight\@empty%
    \LetLtxMacro\elweight\@empty%
    \LetLtxMacro\ulweight\@empty%
%   \LetLtxMacro\itshape\@empty%
%   \LetLtxMacro\scshape\@empty%
%   \LetLtxMacro\upshape\@empty%
    \LetLtxMacro\dfshape\@empty%
    \LetLtxMacro\swshape\@empty%
    \LetLtxMacro\ornament\@gobble%
}

\newrobustcmd{\LWR@HTML@textln}[1]{\InlineClass{textln}{#1}}
\newrobustcmd{\LWR@HTML@textos}[1]{\textsc{#1}}
\newrobustcmd{\LWR@HTML@textin}[1]{#1}
\newrobustcmd{\LWR@HTML@textsu}[1]{#1}
% \newrobustcmd{\LWR@HTML@textsi}[1]{#1}
\newrobustcmd{\LWR@HTML@textdf}[1]{#1}

\ifdef{\LWR@HTML@textsw}{}{% duplicated by fontaxes
    \newrobustcmd{\LWR@HTML@textsw}[1]{#1}
    \LWR@formatted{textsw}
}

\newrobustcmd{\LWR@HTML@textti}[1]{#1}
\newrobustcmd{\LWR@HTML@textlt}[1]{#1}
\newrobustcmd{\LWR@HTML@textof}[1]{#1}
\newrobustcmd{\LWR@HTML@textalt}[1]{#1}
\newrobustcmd{\LWR@HTML@textreg}[1]{#1}
\newrobustcmd{\LWR@HTML@emboss}[1]{#1}
\newrobustcmd{\LWR@HTML@textorn}[1]{#1}
\newrobustcmd{\LWR@HTML@textqt}[1]{#1}
\newrobustcmd{\LWR@HTML@textsh}[1]{#1}
\newrobustcmd{\LWR@HTML@texttm}[1]{\texttt{#1}}
\newrobustcmd{\LWR@HTML@texttv}[1]{\texttt{#1}}
\newrobustcmd{\LWR@HTML@textl}[1]{\InlineClass{textln}{#1}}
```

```
\newrobustcmd{\LWR@HTML@texto}[1]{\textsc{#1}}
\newrobustcmd{\LWR@HTML@textp}[1]{\InlineClass{textp}{#1}}
\newrobustcmd{\LWR@HTML@textt}[1]{\InlineClass{textt}{#1}}
\newrobustcmd{\LWR@HTML@textpl}[1]{#1}
\newrobustcmd{\LWR@HTML@textpo}[1]{\textsc{#1}}
\newrobustcmd{\LWR@HTML@texttl}[1]{\InlineClass{textln}{#1}}
\newrobustcmd{\LWR@HTML@textto}[1]{\textsc{#1}}
\newrobustcmd{\LWR@HTML@textol}[1]{#1}
\newrobustcmd{\LWR@HTML@textswash}[1]{#1}
\newrobustcmd{\LWR@HTML@textu}[1]{#1}
\newrobustcmd{\LWR@HTML@textscu}[1]{#1}
\newrobustcmd{\LWR@HTML@textui}[1]{\LWR@HTML@textit{#1}}
\newrobustcmd{\LWR@HTML@textri}[1]{#1}
\newrobustcmd{\LWR@HTML@textnw}[1]{#1}
\newrobustcmd{\LWR@HTML@textcd}[1]{#1}
\newrobustcmd{\LWR@HTML@textec}[1]{#1}
\newrobustcmd{\LWR@HTML@textuc}[1]{#1}
\newrobustcmd{\LWR@HTML@textet}[1]{#1}
\newrobustcmd{\LWR@HTML@textep}[1]{#1}
\newrobustcmd{\LWR@HTML@textex}[1]{#1}
\newrobustcmd{\LWR@HTML@textux}[1]{#1}
\newrobustcmd{\LWR@HTML@textrw}[1]{#1}
\newrobustcmd{\LWR@HTML@textmb}[1]{{\LWR@HTML@mbweight\InlineClass{textmb}{#1}}}
\newrobustcmd{\LWR@HTML@textdb}[1]{{\LWR@HTML@dbweight\InlineClass{textdb}{#1}}}
\newrobustcmd{\LWR@HTML@textsb}[1]{{\LWR@HTML@sbweight\InlineClass{textsb}{#1}}}
% \newrobustcmd{\LWR@HTML@texteb}[1]}{#1}
\newrobustcmd{\LWR@HTML@textub}[1]{{\LWR@HTML@ubweight\InlineClass{textub}{#1}}}
% \newrobustcmd{\LWR@HTML@textlg}[1]}{#1}
\newrobustcmd{\LWR@HTML@textel}[1]{{\LWR@HTML@elweight\InlineClass{textel}{#1}}}
\newrobustcmd{\LWR@HTML@textul}[1]{{\LWR@HTML@ulweight\InlineClass{textul}{#1}}}

\LWR@formatted{textln}
\LWR@formatted{textos}
\LWR@formatted{textin}
\LWR@formatted{textsu}
% \LWR@formatted{textsi}
\LWR@formatted{textdf}
\LWR@formatted{textti}
\LWR@formatted{textlt}
\LWR@formatted{textof}
\LWR@formatted{textalt}
\LWR@formatted{textreg}
\LWR@formatted{emboss}
\LWR@formatted{textorn}
\LWR@formatted{textqt}
\LWR@formatted{textsh}
\LWR@formatted{texttm}
\LWR@formatted{texttv}
\LWR@formatted{textl}
\LWR@formatted{texto}
\LWR@formatted{textp}
\LWR@formatted{textt}
\LWR@formatted{textpl}
\LWR@formatted{textpo}
\LWR@formatted{texttl}
\LWR@formatted{textto}
\LWR@formatted{textol}
\LWR@formatted{textswash}
\LWR@formatted{textu}
\LWR@formatted{textscu}
```

```
\LWR@formatted{textui}
\LWR@formatted{textri}
\LWR@formatted{textnw}
\LWR@formatted{textcd}
\LWR@formatted{textec}
\LWR@formatted{textuc}
\LWR@formatted{textet}
\LWR@formatted{textep}
\LWR@formatted{textex}
\LWR@formatted{textux}
\LWR@formatted{textrw}
\LWR@formatted{textmb}
\LWR@formatted{textdb}
\LWR@formatted{textsb}
% \LWR@formatted{texteb}
\LWR@formatted{textub}
% \LWR@formatted{textlg}
\LWR@formatted{textel}
\LWR@formatted{textul}


\FilenameNullify{%
    \LetLtxMacro\textln\@firstofone%
    \LetLtxMacro\textos\@firstofone%
    \LetLtxMacro\textin\@firstofone%
    \LetLtxMacro\textsu\@firstofone%
%   \LetLtxMacro\textsi\@firstofone%
    \LetLtxMacro\textdf\@firstofone%
    \LetLtxMacro\textsw\@firstofone%
    \LetLtxMacro\textti\@firstofone%
    \LetLtxMacro\textlt\@firstofone%
    \LetLtxMacro\textof\@firstofone%
    \LetLtxMacro\textalt\@firstofone%
    \LetLtxMacro\textreg\@firstofone%
    \LetLtxMacro\emboss\@firstofone%
    \LetLtxMacro\textorn\@firstofone%
    \LetLtxMacro\textqt\@firstofone%
    \LetLtxMacro\textsh\@firstofone%
    \LetLtxMacro\texttm\@firstofone%
    \LetLtxMacro\texttv\@firstofone%
    \LetLtxMacro\textl\@firstofone%
    \LetLtxMacro\texto\@firstofone%
    \LetLtxMacro\textp\@firstofone%
    \LetLtxMacro\textt\@firstofone%
    \LetLtxMacro\textpl\@firstofone%
    \LetLtxMacro\textpo\@firstofone%
    \LetLtxMacro\texttl\@firstofone%
    \LetLtxMacro\textto\@firstofone%
    \LetLtxMacro\textol\@firstofone%
    \LetLtxMacro\textswash\@firstofone%
    \LetLtxMacro\textu\@firstofone%
    \LetLtxMacro\textscu\@firstofone%
    \LetLtxMacro\textui\@firstofone%
    \LetLtxMacro\textri\@firstofone%
    \LetLtxMacro\textnw\@firstofone%
    \LetLtxMacro\textcd\@firstofone%
    \LetLtxMacro\textec\@firstofone%
    \LetLtxMacro\textuc\@firstofone%
    \LetLtxMacro\textet\@firstofone%
    \LetLtxMacro\textep\@firstofone%
```

```
    \LetLtxMacro\textex\@firstofone%
    \LetLtxMacro\textux\@firstofone%
    \LetLtxMacro\textrw\@firstofone%
    \LetLtxMacro\textmb\@firstofone%
    \LetLtxMacro\textdb\@firstofone%
    \LetLtxMacro\textsb\@firstofone%
%   \LetLtxMacro\texteb\@firstofone%
    \LetLtxMacro\textub\@firstofone%
%   \LetLtxMacro\textlg\@firstofone%
    \LetLtxMacro\textel\@firstofone%
    \LetLtxMacro\textul\@firstofone%
}

\providecommand*{\zeroslash}{0}
\newrobustcmd*{\LWR@HTML@zeroslash}{0}
\LWR@formatted{zeroslash}
```

File 336 **lwarp-nicefrac.sty**

§ 445 Package nicefrac

(*Emulates or patches code by* Axel Reichert.)

nicefrac (*Pkg*) nicefrac is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{nicefrac}[1998/08/04]

\DeclareRobustCommand*{\LWR@HTML@@UnitsNiceFrac}[3][]{%
    {% localize font selection
        #1{%
            \LWR@textcurrentfont{%
                \InlineClass{numerator}{#2}%
                /%
                \InlineClass{denominator}{#3}%
            }%
        }%
    }%
}

\LWR@formatted{@UnitsNiceFrac}

\DeclareRobustCommand*{\LWR@HTML@@UnitsUglyFrac}[3][]{%
    {% localize font selection
        #1{\LWR@textcurrentfont{#2/#3}}%
    }%
}

\LWR@formatted{@UnitsUglyFrac}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\nicefrac}[3][]{\mathinner{{}^{#2}\!/\!_{#3}}}}
\end{warpMathJax}
```

File 337 **lwarp-niceframe.sty**

§ 446 Package niceframe

`niceframe` (*Pkg*) niceframe is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{niceframe}% the original date is in yyyy/dd/mm format

\newcommand{\LWR@niceframe}[3]{%
    \begin{LWR@setvirtualpage}*%
    \setlength{\LWR@templengthone}{#1}%
    \begin{BlockClass}[max-width:\LWR@printlength{\LWR@templengthone}]{#3}%
    #2
    \end{BlockClass}%
    \end{LWR@setvirtualpage}%
}

\newcommand{\niceframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{niceframe}}
\newcommand{\curlyframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{curlyframe}}
\newcommand{\artdecoframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{artdecoframe}}

\newcommand{\generalframe}[9]{\LWR@niceframe{\textwidth}{#9}{generalframe}}
```

File 338 **lwarp-nicematrix.sty**

§ 447 Package nicematrix

(*Emulates or patches code by* F. Pantigny.)

`nicematrix` (*Pkg*) nicematrix is used as-is for SVG math, and is emulated for MathJax.

⚠ MathJax Keys/values are ignored in MathJax. `\Cdots`, etc. do not span multiple cells. `AutoNiceMatrix`, etc. are not supported for MathJax. SVG math output preserves all nicematrix features. To force SVG output for one or more consecutive math expressions, for inline math use `\inlinemathother` and `\inlinemathnormal`, or for display math use `\displaymathother` and `\displaymathnormal`.

for HTML output: Skip the test for `array`, which does not work with lwarp:

```
\PassOptionsToPackage{no-test-for-array}{nicematrix}

\LWR@ProvidesPackagePass{nicematrix}[2022/10/06]
```

`NiceTabular` must be converted to SVG to support the various nicematrix options:

```
\begin{warpHTML}
\BeforeBeginEnvironment{NiceTabular}{%
    \begin{lateximage}[-nicematrix-~\PackageDiagramAltText]%
}
\AfterEndEnvironment{NiceTabular}{\end{lateximage}}
\BeforeBeginEnvironment{NiceTabular*}{%
```

```
    \begin{lateximage}[-nicematrix-~\PackageDiagramAltText]%
}
\AfterEndEnvironment{NiceTabular*}{\end{lateximage}}
\end{warpHTML}
```

Special handling for the optional arguments, and the lack of a delimiter:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\LWRnicearrayarray}[1]{\begin{array}{#1}}}
\CustomizeMathJax{\def\LWRnicearrayarrayopt#1[#2] {\begin{array}{#1}}}

\CustomizeMathJax{%
    \newenvironment{NiceArray}[2][]%
        {\ifnextchar[{\LWRnicearrayarrayopt{#2}}{\LWRnicearrayarray{#2}}}%
        {\end{array}}%
}

\CustomizeMathJax{%
    \newcommand{\LWRnicearraywithdelimtwo}[2][]{%
        \ifnextchar[{\LWRnicearrayarrayopt{#2}}{\LWRnicearrayarray{#2}}%
    }%
}
```

General case with left / right delimiters:

```
\CustomizeMathJax{%
    \newenvironment{NiceArrayWithDelims}[2]%
        {%
            \def\LWRnicearrayrightdelim{\right#2}%
            \left#1%
            \LWRnicearraywithdelimtwo%
        }%
        {\end{array}\LWRnicearrayrightdelim}%
}
```

Instances of specific delimiters:

```
\CustomizeMathJax{%
    \newenvironment{pNiceArray}
        {\begin{NiceArrayWithDelims}{(}{)}}
        {\end{NiceArrayWithDelims}}
}

\CustomizeMathJax{%
    \newenvironment{bNiceArray}
        {\begin{NiceArrayWithDelims}{[}{]}}
        {\end{NiceArrayWithDelims}}
}

\CustomizeMathJax{%
    \newenvironment{BNiceArray}
        {\begin{NiceArrayWithDelims}{\{}{\}}}
        {\end{NiceArrayWithDelims}}
}

\CustomizeMathJax{%
    \newenvironment{vNiceArray}
        {\begin{NiceArrayWithDelims}{\vert}{\vert}}
        {\end{NiceArrayWithDelims}}
}
```

```
60
61 \CustomizeMathJax{%
62     \newenvironment{VNiceArray}
63         {\begin{NiceArrayWithDelims}{\Vert}{\Vert}}
64         {\end{NiceArrayWithDelims}}
65 }
```

Ignore optional arg and use standard environments:

```
66 \CustomizeMathJax{\newenvironment{NiceMatrix}[1][]{\begin{matrix}}{\end{matrix}}}
67 \CustomizeMathJax{\newenvironment{pNiceMatrix}[1][]{\begin{pmatrix}}{\end{pmatrix}}}
68 \CustomizeMathJax{\newenvironment{bNiceMatrix}[1][]{\begin{bmatrix}}{\end{bmatrix}}}
69 \CustomizeMathJax{\newenvironment{BNiceMatrix}[1][]{\begin{Bmatrix}}{\end{Bmatrix}}}
70 \CustomizeMathJax{\newenvironment{vNiceMatrix}[1][]{\begin{vmatrix}}{\end{vmatrix}}}
71 \CustomizeMathJax{\newenvironment{VNiceMatrix}[1][]{\begin{Vmatrix}}{\end{Vmatrix}}}
```

Ignore optional argument and size. Print contents.

```
72 \CustomizeMathJax{\newcommand{\LWRnicematrixBlock}[1]{#1}}
73 \CustomizeMathJax{\def\LWRnicematrixBlockopt<#1>#2{#2}}
74
75 \CustomizeMathJax{%
76   \newcommand{\Block}[2][]{\ifnextchar<\LWRnicematrixBlockopt\LWRnicematrixBlock}%
77 }
```

Form an approximation:

```
78 \CustomizeMathJax{%
79     \newcommand{\diagbox}[2]{%
80         \begin{array}{l}\hfill\quad#2\\\hline#1\quad\hfill\end{array}%
81     }%
82 }
```

More approximations:

```
83 \CustomizeMathJax{\let\hdottedline\hdashline}


84 \CustomizeMathJax{\newcommand{\Hline}[1][]{\hline}}


85 \CustomizeMathJax{\newcommand{\CodeBefore}{}}
86 \CustomizeMathJax{\newcommand{\Body}{}}
87 \CustomizeMathJax{\newcommand{\CodeAfter}{}}
88 \CustomizeMathJax{\newcommand{\line}[3][]{}}
89 \CustomizeMathJax{\newcommand{\RowStyle}[2][]{}}
90 \CustomizeMathJax{\newcommand{\LWRSubMatrix}[1][]{}}
91 \CustomizeMathJax{\newcommand{\SubMatrix}[4]{\LWRSubMatrix}}
92 \CustomizeMathJax{\newcommand{\OverBrace}[4][]{}}
93 \CustomizeMathJax{\newcommand{\UnderBrace}[4][]{}}
94 \CustomizeMathJax{\newcommand{\ShowCellNames}{}}


95 \CustomizeMathJax{\newcommand{\cellcolor}[3][]{}}
96 \CustomizeMathJax{\newcommand{\rowcolor}[3][]{}}
97 \CustomizeMathJax{\newcommand{\LWRrowcolors}[1][]{}}
98 \CustomizeMathJax{\newcommand{\rowcolors}[4][]{\LWRrowcolors}}
99 \CustomizeMathJax{\newcommand{\rowlistcolors}[3][]{\LWRrowcolors}}
100 \CustomizeMathJax{\newcommand{\columncolor}[3][]{}}
101 \CustomizeMathJax{\newcommand{\rectanglecolor}[4][]{}}
```

```
\CustomizeMathJax{\newcommand{\arraycolor}[2][]{}}
\CustomizeMathJax{\newcommand{\chessboardcolors}[3][]{}}

\CustomizeMathJax{\newcommand{\ldots}[1][]{\dots}}
\CustomizeMathJax{\newcommand{\Cdots}[1][]{\cdots}}
\CustomizeMathJax{\newcommand{\Vdots}[1][]{\vdots}}
\CustomizeMathJax{\newcommand{\Ddots}[1][]{\ddots}}
\CustomizeMathJax{\newcommand{\Iddots}[1][]{\mathinner{\unicode{x22F0}}}}

\CustomizeMathJax{\newcommand{\Hdotsfor}[1]{\ldots}}
\CustomizeMathJax{\newcommand{\Vdotsfor}[1]{\vdots}}
```

There is no way to emulate AutoNiceMatrix in MathJax.

```
\CustomizeMathJax{\newcommand{\AutoNiceMatrix}[2]{\text{(AutoNiceMatrix #1)}}}
\CustomizeMathJax{\let\pAutoNiceMatrix\AutoNiceMatrix}
\CustomizeMathJax{\let\bAutoNiceMatrix\AutoNiceMatrix}
\CustomizeMathJax{\let\BAutoNiceMatrix\AutoNiceMatrix}
\CustomizeMathJax{\let\vAutoNiceMatrix\AutoNiceMatrix}
\CustomizeMathJax{\let\VAutoNiceMatrix\AutoNiceMatrix}
\end{warpMathJax}
```

File 339 **lwarp-noitcrul.sty**

§ 448 Package noitcrul

(*Emulates or patches code by* Paul Ebermann.)

noitcrul (*Pkg*) noitcrul is used as-is for SVG and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{noitcrul}[2006/04/11]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\noitUnderline}[1]{\underline{#1}\!}}
\end{warpMathJax}
```

File 340 **lwarp-nolbreaks.sty**

§ 449 Package nolbreaks

(*Emulates or patches code by* Donald Arseneau.)

nolbreaks (*Pkg*) nolbreaks is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{nolbreaks}[2012/05/31]

\NewDocumentCommand{\nolbreaks}{s m}{\InlineClass{nolbreaks}{#2}}
```

File 341 **lwarp-nomencl.sty**

§ 450 Package nomencl

(*Emulates or patches code by* Boris Veytsman, Bernd Schandl, Lee Netherton, CV Radhakrishnan.)

nomencl (*Pkg*) nomencl is patched for use by lwarp.

To process the HTML nomenclature:

```
makeindex <project>_html.nlo -s nomencl.ist -o <project>_html.nls
```

for HTML output:

```
\LWR@ProvidesPackagePass{nomencl}[2005/09/22]
```

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\def\@@@nomenclature[#1]#2#3{%
 \def\@tempa{#2}\def\@tempb{#3}%
 \protected@write\@nomenclaturefile{}%
  {\string\nomenclatureentry{#1\nom@verb\@tempa @[{\nom@verb\@tempa}]%
      \begingroup\nom@verb\@tempb\protect\nomeqref{\theequation}%
        |nompageref}{\theLWR@previousautopagelabel}}%   lwarp
 \endgroup
 \@esphack}

\renewcommand*{\pagedeclaration}[1]{, \nameref{\BaseJobname-autopage-#1}}%
```

File 342 **lwarp-nonfloat.sty**

§ 451 Package nonfloat

(*Emulates or patches code by* Kai Rascher.)

nonfloat (*Pkg*) nonfloat is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{nonfloat}[1999/07/05]
```

```
\LetLtxMacro\topcaption\caption
\newcommand{\figcaption}{\def\@captype{figure}\caption}
\newcommand{\tabcaption}{\def\@captype{table}\topcaption}
\newenvironment{narrow}[2]{}{}
```

File 343 **lwarp-nonumonpart.sty**

§ 452 Package nonumonpart

nonumonpart (*Pkg*) nonumonpart is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nonumonpart}[2011/04/15]
```

File 344 **lwarp-nopageno.sty**

§ 453 Package nopageno

nopageno (*Pkg*) nopageno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nopageno}[1989/01/01]
```

File 345 **lwarp-notes.sty**

§ 454 Package notes

notes (*Pkg*) notes is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{notes}[2002/10/29]

\newcommand*{\LWR@notes@onenote}[2]{%
\newenvironment{#1}
    {
        \BlockClass{notes#1}
        \begin{BlockClass}{notesicon}\textcircled{~#2~}\end{BlockClass}
        \BlockClass{notescontents}
    }
    {\endBlockClass\endBlockClass}
}

\LWR@notes@onenote{importantnote}{!}

\LWR@notes@onenote{warningnote}{--}

\LWR@notes@onenote{informationnote}{i}
```

File 346 **lwarp-notespages.sty**

§ 455 Package notespages

notespages (*Pkg*) notespages is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{notespages}[2016/08/21]

\newcommand*{\npnotesname}{}
\newcommand*{\npnotestext}{}
\newcommand*{\remainingtextheight}{}
\newdimen\remainingtextheight
\newcommand*{\notestitletext}{}
\newcommand*{\notesareatext}{}
\newcommand*{\npnpinfo}[1]{}
\newcommand*{\tracingnpmarks}{}
\newcommand*{\notespage}[1][]{}
\newcommand*{\notespages}[1][]{}
\newcommand*{\notesfill}[1][]{}
\newcommand*{\setnotespages}[1]{}
\newcommand*{\definenotesoption}[2]{}
\newcommand{\definenotesstyle}[2]{}
\newcommand{\definetitlestyle}[2]{}
\newcommand{\nppatchchapter}[1]{}
\newcommand{\npunpatchchapter}{}
```

File 347 **lwarp-nowidow.sty**

§ 456 Package nowidow

(*Emulates or patches code by* Raphaël Pinson.)

nowidow (*Pkg*) nowidow is ignored.

for HTML output: Discard all options for lwarp-nowidow.

```
\ProvidesPackage{lwarp-nowidow}[2011/09/20]
```

\nowidow [⟨*lines*⟩]

\setnowidow [⟨*lines*⟩]

```
\newcommand*{\nowidow}[1][]{}
\newcommand*{\setnowidow}[1][]{}
```

\noclub [⟨*lines*⟩]

\setnoclub [⟨*lines*⟩]

```
\newcommand*{\noclub}[1][]{}
\newcommand*{\setnoclub}[1][]{}
```

File 348 **lwarp-ntheorem.sty**

§ 457 Package ntheorem

(*Emulates or patches code by* Wolfgang May, Andreas Schedler.)

ntheorem (*Pkg*) ntheorem is patched for use by lwarp.

Table 20: Ntheorem package — css styling of theorems and proofs

Theorem: of class theorembody<theoremstyle>

Theorem Header: of class theoremheader<style>

where <theoremstyle> is plain, break, etc.

§ 457.1 Limitations

⚠ Font control This conversion is not total. Font control is via css, and the custom LaTeX font settings are ignored.

⚠ Equation numbering ntheorem has a bug with equation numbering in $\mathcal{AMS}$ environments when the option thref is used. lwarp does not share this bug, so equations with \split, etc, are numbered correctly with lwarp's HTML output, but not with the print output. It is recommended to use cleveref instead of ntheorem's thref option.

§ 457.2 Options

Options `amsthm` or `standard` choose which set of theorems and proofs to initialize.

⚠ Disabled options

The options `thmmarks` and `amsmath` are disabled, since they heavily modify the underlying math code. Theorem marks are emulated. The AMS-math modifications are not done.

Option `thref` is disabled because cleveref functions are used instead. `\thref` is emulated.

Option `hyperref` is disabled because lwarp emulated hyperref.

for HTML output:

Some disabled options:

```
\DeclareOption{thref}{
    \AtEndDocument{
        \PackageWarningNoLine{lwarp}{%
            Lwarp uses cleveref, which takes over ntheorem's\MessageBreak
            referencing, including
                \protect\label \space and \protect\thref.\MessageBreak
            Cleveref does not accept ntheorem's optional\MessageBreak
            argument for \protect\label, so it will appear\MessageBreak
            in the text.  It is recommended to remove the\MessageBreak
           thref option, \protect\usepackage{cleveref} instead,\MessageBreak
            and remove any trailing optional arguments for \protect\label%
        }%
    }
}


\newbool{LWR@ntheoremmarks}
\boolfalse{LWR@ntheoremmarks}

\DeclareOption{thmmarks}{
\booltrue{LWR@ntheoremmarks}
\newif\ifsetendmark\setendmarktrue
}


\newbool{LWR@ntheoremamsthm}
\boolfalse{LWR@ntheoremamsthm}

\DeclareOption{amsthm}{\booltrue{LWR@ntheoremamsthm}}


\DeclareOption{amsmath}{}
\DeclareOption{hyperref}{}

\LWR@ProvidesPackagePass{ntheorem}[2011/08/15]
```

§ 457.3 Remembering the theorem style

Storage for the style being used for new theorems.

```
\newcommand{\LWR@newtheoremstyle}{plain}

\AtBeginDocument{
\IfPackageLoadedTF{cleveref}{
\gdef\@thm#1#2#3{%
   \if@thmmarks
     \stepcounter{end\InTheoType ctr}%
   \fi
   \renewcommand{\InTheoType}{#1}%
   \if@thmmarks
     \stepcounter{curr#1ctr}%
     \setcounter{end#1ctr}{0}%
   \fi
   \refstepcounter[#1]{#2}%  <<< cleveref modification
   \theorem@prework
    \LWR@forcenewpage% lwarp

    \LWR@printpendingfootnotes%                  lwarp

    \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
   \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
   \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
     \ifthm@inframe
       \thm@topsep\theoreminframepreskipamount
       \thm@topsepadd\theoreminframepostskipamount
     \else
       \thm@topsep\theorempreskipamount
       \thm@topsepadd\theorempostskipamount
     \fi
    \else% oldframeskips
      \thm@topsep\theorempreskipamount
      \thm@topsepadd \theorempostskipamount
      \ifvmode\advance\thm@topsepadd\partopsep\fi
   \fi
   \@topsep\thm@topsep
   \@topsepadd\thm@topsepadd
   \advance\linewidth -\theorem@indent
   \advance\linewidth -\theorem@rightindent
   \advance\@totalleftmargin \theorem@indent
   \parshape \@ne \@totalleftmargin \linewidth
   \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
}
}{% not @ifpackageloaded{cleveref}
\gdef\@thm#1#2#3{%
   \if@thmmarks
     \stepcounter{end\InTheoType ctr}%
   \fi
   \renewcommand{\InTheoType}{#1}%
   \if@thmmarks
     \stepcounter{curr#1ctr}%
     \setcounter{end#1ctr}{0}%
   \fi
   \refstepcounter{#2}%
   \theorem@prework
    \LWR@forcenewpage% lwarp
```

```
    \LWR@printpendingfootnotes%                   lwarp

    \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
  \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
  \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
    \ifthm@inframe
      \thm@topsep\theoreminframepreskipamount
      \thm@topsepadd\theoreminframepostskipamount
    \else
      \thm@topsep\theorempreskipamount
      \thm@topsepadd\theorempostskipamount
    \fi
   \else% oldframeskips
     \thm@topsep\theorempreskipamount
     \thm@topsepadd \theorempostskipamount
     \ifvmode\advance\thm@topsepadd\partopsep\fi
  \fi
  \@topsep\thm@topsep
  \@topsepadd\thm@topsepadd
  \advance\linewidth -\theorem@indent
  \advance\linewidth -\theorem@rightindent
  \advance\@totalleftmargin \theorem@indent
  \parshape \@ne \@totalleftmargin \linewidth
  \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
}
}
}% AtBeginDocument
```

Patched to remember the style being used for new theorems:

```
\gdef\theoremstyle#1{%
  \@ifundefined{th@#1}{\@warning
         {Unknown theoremstyle '#1'. Using 'plain'}%
         \theorem@style{plain}
           \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
           }%
     {
       \theorem@style{#1}
       \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
       }
}
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```

\gdef\@xnthm#1#2[#3]{%
 \ifthm@tempif
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \expandafter\@ifundefined{c@#1}%
       {\@definecounter{#1}}{}%
    \@newctr{#1}[#3]%
    \expandafter\xdef\csname the#1\endcsname{%
      \expandafter\noexpand\csname the#3\endcsname \@thmcountersep
         {\noexpand\csname\the\theoremnumbering\endcsname{#1}}}%
    \expandafter\gdef\csname mkheader@#1\endcsname
      {\csname setparms@#1\endcsname
       \@thm{#1}{#1}{#2}
       }%
```

```
    \global\@namedef{end#1}{\@endtheorem}
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
  \fi
}

\gdef\@ynthm#1#2{%
  \ifthm@tempif
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \expandafter\@ifundefined{c@#1}%
       {\@definecounter{#1}}{}%
    \expandafter\xdef\csname the#1\endcsname
       {\noexpand\csname\the\theoremnumbering\endcsname{#1}}%
    \expandafter\gdef\csname mkheader@#1\endcsname
      {\csname setparms@#1\endcsname
       \@thm{#1}{#1}{#2}
       }%
    \global\@namedef{end#1}{\@endtheorem}
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
  \fi
}

\gdef\@othm#1[#2]#3{%
  \@ifundefined{c@#2}{\@nocounterr{#2}}%
   {\ifthm@tempif
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \global\@namedef{the#1}{\@nameuse{the#2}}%
    \expandafter\protected@xdef\csname num@addtheoremline#1\endcsname{%
              \noexpand\@num@addtheoremline{#1}{#3}}%
    \expandafter\protected@xdef\csname nonum@addtheoremline#1\endcsname{%
              \noexpand\@nonum@addtheoremline{#1}{#3}}%
    \theoremkeyword{#3}%
    \expandafter\protected@xdef\csname #1Keyword\endcsname
             {\the\theoremkeyword}%
    \expandafter\gdef\csname mkheader@#1\endcsname
      {\csname setparms@#1\endcsname
               \@thm{#1}{#2}{#3}
       }%
    \global\@namedef{end#1}{\@endtheorem}
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
  \fi}
}
```

§ 457.4 HTML cross-referencing

Mimics a float by incrementing the float counter and generating an HTML anchor. These are used for list-of-theorem cross-references.

```
\newcommand{\LWR@inctheorem}{%
\addtocounter{LWR@thisautoid}{1}%
\LWR@stoppars%
\LWR@htmltag{%
  a id=\textquotedbl\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}\textquotedbl%
}%
\LWR@htmltag{/a}\LWR@orignewline%
\LWR@startpars%
}
```

§ 457.5 \newtheoremstyle

The following are patched for css.

These were in individual files `thp.sty` for plain, `thmb.sty` for margin break, etc. They are gathered together here.

Each theorem is encased in a `BlockClass` environment of class `theorembody<style>`.

Each header is encased in an `\InlineClass` of class `theoremheader<style>`.

```
\gdef\newtheoremstyle#1#2#3{%
  \expandafter\@ifundefined{th@#1}%
   {\expandafter\gdef\csname th@#1\endcsname{%
    \def\@begintheorem####1####2{%
    \LWR@inctheorem% lwarp
    #2}%
    \def\@opargbegintheorem####1####2####3{%
    \LWR@inctheorem% lwarp
    #3}%
}%
}%
{\PackageError{\basename}{Theorem style #1 already defined}\@eha}
}
```

§ 457.6 Standard styles

```
\renewtheoremstyle{plain}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}

\renewtheoremstyle{break}%
  {\item[
    \InlineClass{theoremheaderbreak}{##1\ ##2\theorem@separator}\newline
    ]}%
  {\item[
    \InlineClass{theoremheaderbreak}%
        {##1\ ##2\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{change}%
  {\item[
    \InlineClass{theoremheaderchange}{##2\ ##1\theorem@separator}]}%
  {\item[
    \InlineClass{theoremheaderchange}{##2\ ##1\ (##3)\theorem@separator}]}

\renewtheoremstyle{changebreak}%
  {\item[
        \InlineClass{theoremheaderchangebreak}%
            {##2\ ##1\theorem@separator}\newline
    ]}%
  {\item[
        \InlineClass{theoremheaderchangebreak}%
            {##2\ ##1\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{margin}%
  {\item[
```

```
        \InlineClass{theoremheadermargin}{##2 \qquad ##1\theorem@separator}
    ]}%
  {\item[
     \InlineClass{theoremheadermargin}{##2 \qquad ##1\ (##3)\theorem@separator}
    ]}

\renewtheoremstyle{marginbreak}%
  {\item[
    \InlineClass{theoremheadermarginbreak}%
        {##2 \qquad ##1\theorem@separator}\newline
    ]}%
  {\item[
    \InlineClass{theoremheadermarginbreak}%
        {##2 \qquad ##1\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{nonumberplain}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\theorem@separator}]}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\ (##3)\theorem@separator}]}

\renewtheoremstyle{nonumberbreak}%
  {\item[
    \InlineClass{theoremheaderbreak}{##1\theorem@separator}\newline
    ]}%
  {\item[
    \InlineClass{theoremheaderbreak}{##1\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{empty}%
  {\item[]}%
  {\item[
    \InlineClass{theoremheaderplain}{##3}]}

\renewtheoremstyle{emptybreak}%
  {\item[]}%
  {\item[
    \InlineClass{theoremheaderplain}{##3}] \ \newline}
```

§ 457.7 Additional objects

The following manually adjust the CSS for the standard configuration objects which are not a purely plain style:

```
\ifbool{LWR@ntheoremamsthm}{}{%
```

Upright text via CSS:

```
    \newtheoremstyle{plainupright}%
    {\item[
        \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
    {\item[
       \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
```

Upright text and small caps header via CSS:

```
    \newtheoremstyle{nonumberplainuprightsc}%
    {\item[
```

```
        \InlineClass{theoremheadersc}{##1\theorem@separator}]}%
    {\item[
        \InlineClass{theoremheadersc}{##1\ (##3)\theorem@separator}]}
}% not amsthm
```

§ 457.8 Renewed standard configuration

The following standard configuration is renewed using the new CSS:

```
\ifbool{LWR@ntheoremamsthm}{}{%

\ifx\thm@usestd\@undefined
\else
    \theoremnumbering{arabic}
    \theoremstyle{plain}
    \RequirePackage{latexsym}
    \theoremsymbol{\Box}
    \theorembodyfont{\itshape}
    \theoremheaderfont{\normalfont\bfseries}
    \theoremseparator{}
    \renewtheorem{Theorem}{Theorem}
    \renewtheorem{theorem}{Theorem}
    \renewtheorem{Satz}{Satz}
    \renewtheorem{satz}{Satz}
    \renewtheorem{Proposition}{Proposition}
    \renewtheorem{proposition}{Proposition}
    \renewtheorem{Lemma}{Lemma}
    \renewtheorem{lemma}{Lemma}
    \renewtheorem{Korollar}{Korollar}
    \renewtheorem{korollar}{Korollar}
    \renewtheorem{Corollary}{Corollary}
    \renewtheorem{corollary}{Corollary}

    \theoremstyle{plainupright}
    \theorembodyfont{\upshape}
    \theoremsymbol{\HTMLunicode{25A1}}% UTF-8 white box
    \renewtheorem{Example}{Example}
    \renewtheorem{example}{Example}
    \renewtheorem{Beispiel}{Beispiel}
    \renewtheorem{beispiel}{Beispiel}
    \renewtheorem{Bemerkung}{Bemerkung}
    \renewtheorem{bemerkung}{Bemerkung}
    \renewtheorem{Anmerkung}{Anmerkung}
    \renewtheorem{anmerkung}{Anmerkung}
    \renewtheorem{Remark}{Remark}
    \renewtheorem{remark}{Remark}
    \renewtheorem{Definition}{Definition}
    \renewtheorem{definition}{Definition}

    \theoremstyle{nonumberplainuprightsc}
    \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
    \renewtheorem{Proof}{Proof}
    \renewtheorem{proof}{Proof}
    \renewtheorem{Beweis}{Beweis}
    \renewtheorem{beweis}{Beweis}
    \qedsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof

    \theoremsymbol{}
\fi
}% not amsthm
```

§ 457.9 amsthm option

Only if the amsthm option was given:

```
\ifbool{LWR@ntheoremamsthm}{

\gdef\th@plain{%
  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderplain}{##1\ ##2.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderplain}{##1\ ##2\ (##3).}
        ]}}

\gdef\th@nonumberplain{%
  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderplain}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderplain}{##1\ (##3).}
        ]}}

\gdef\th@definition{%
  \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderdefinition}{##1\ ##2.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderdefinition}{##1\ ##2\ (##3).}
        ]}}

\gdef\th@nonumberdefinition{%
  \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderdefinition}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderdefinition}{##1\ (##3).}
        ]}}

\gdef\th@remark{%
  \def\theorem@headerfont{\itshape}\normalfont%
```

```
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderremark}{##1\ ##2.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderremark}{##1\ ##2\ (##3).}
        ]}}

\gdef\th@nonumberremark{%
  \def\theorem@headerfont{\itshape}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderremark}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderremark}{##1\ (##3).}
        ]}}

\gdef\th@proof{%
  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderproof}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderproof}{##1\ (##3).}
        ]}}



\newcounter{proof}%
\if@thmmarks
    \newcounter{currproofctr}%
    \newcounter{endproofctr}%
\fi

\gdef\proofSymbol{\openbox}

\newcommand{\proofname}{Proof}

\newenvironment{proof}[1][\proofname]{
    \th@proof
    \def\theorem@headerfont{\itshape}%
    \normalfont
    \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
    \@thm{proof}{proof}{#1}
}%
{\@endtheorem}

}{}% amsthm option
```

§ 457.10 Ending a theorem

Patched for css:

```
\let\LWR@origendtheorem\@endtheorem
\renewcommand{\@endtheorem}{%
\ifbool{LWR@ntheoremmarks}{%
    \ifsetendmark%
    \InlineClass{theoremendmark}{\csname\InTheoType Symbol\endcsname}%
    \setendmarkfalse%
    \fi%
}{}%
\LWR@origendtheorem% also does \@endtrivlist
\ifbool{LWR@ntheoremmarks}{\global\setendmarktrue}{}%

    \LWR@printpendingfootnotes%                    lwarp

\endBlockClass%
}
```

§ 457.11 \NoEndMark

```
\gdef\NoEndMark{\global\setendmarkfalse}
```

§ 457.12 List-of

Redefined to reuse the float mechanism to add list-of-theorem links:

\thm@thmline {⟨*1: printed type*⟩} {⟨*2: #*⟩} {⟨*3: optional*⟩} {⟨*4: page*⟩}

```
\renewcommand{\thm@@thmline@noname}[4]{%
\hypertocfloat{1}{theorem}{thm}{#2 #3}{}%
}

\renewcommand{\thm@@thmline@name}[4]{%
\hypertocfloat{1}{theorem}{thm}{#1 #2 #3}{}%
}
```

This was redefined by ntheorem when loaded, so it is now redefined for lwarp:

```
\def\thm@@thmline{\thm@@thmline@name}
```

Patch for css:

```
\def\listtheorems#1{
\LWR@htmlelementclass{nav}{lothm}%
\begingroup
\c@tocdepth=-2%
\def\thm@list{#1}\thm@processlist
\endgroup
\LWR@htmlelementclassend{nav}{lothm}%
}
```

§ 457.13 Symbols

Proof QED symbol:

```
\newcommand{\qed}{\qquad\the\qedsymbol}

\AtBeginDocument{
\@ifundefined{LWR@orig@openbox}{
\LetLtxMacro\LWR@orig@openbox\openbox
\LetLtxMacro\LWR@orig@blacksquare\blacksquare
\LetLtxMacro\LWR@orig@Box\Box

\def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
\def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
\def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\openbox\LWR@orig@openbox%
\LetLtxMacro\blacksquare\LWR@orig@blacksquare%
\LetLtxMacro\Box\LWR@orig@Box%
}% appto
}{}% @ifundefined
}% AtBeginDocument
```

§ 457.14 Cross-referencing

`\thref {`⟨*label*⟩`}`

```
\newcommand*{\thref}[1]{\cref{#1}}%
```

File 349 **lwarp-octave.sty**

§ 458 Package octave

(*Emulates or patches code by* Andrew A. Cashner.)

octave (*Pkg*) octave is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{octave}[2017/10/31]
```

Remove the leading 1pt kern:

```
\RenewDocumentCommand{\@PrintTicks}{ m }{%
\kern-1pt%  lwarp
\@TickNum = #1%
\loop
\@Tick{}%
\advance\@TickNum by -1
\ifnum\@TickNum > 0
\repeat
}
```

Use unicode for the prime character:

```
\RenewDocumentCommand{\@Tick}{}{\HTMLunicode{2032}}
```

Catch the inline font:

```
\RenewDocumentCommand{\pitch}{ m o m }{%
\if@OctaveNumber%
{%
    \pitchfont{%
        \LWR@textcurrentfont{%  lwarp
            \MakeUppercase{#1}%
            \IfValueTF{#2}{#2}{}\textsubscript{#3}%
        }%
    }%
}%
\else%
{%
    \pitchfont{%
        \LWR@textcurrentfont{%  lwarp
            \@GetOctaveTick{#1}[#2]{#3}%
        }%
    }%
}%
\fi%
}
```

The original was hard to adapt to lwarp's handling of &.

```
\StartDefiningTabulars
\renewcommand{\octavetable}{%
\begin{tabular}{ll}
\octaveprimes \pitch{C}{0} & \octavenumbers \pitch{C}{0} \\
\octaveprimes \pitch{C}{1} & \octavenumbers \pitch{C}{1} \\
\octaveprimes \pitch{C}{2} & \octavenumbers \pitch{C}{2} \\
\octaveprimes \pitch{C}{3} & \octavenumbers \pitch{C}{3} \\
\octaveprimes \pitch{C}{4} & \octavenumbers \pitch{C}{4} \\
\octaveprimes \pitch{C}{5} & \octavenumbers \pitch{C}{5} \\
\octaveprimes \pitch{C}{6} & \octavenumbers \pitch{C}{6} \\
\octaveprimes \pitch{C}{7} & \octavenumbers \pitch{C}{7} \\
\end{tabular}
}
\StopDefiningTabulars
```

File 350 **lwarp-orcidlink.sty**

§ 459 Package

orcidlink

(*Emulates or patches code by* Leo C. Stein.)

orcidlink (*Pkg*) orcidlink is patched for use by lwarp.

for HTML output:

```
\RequirePackage{lwarp-scalerel}

\LWR@ProvidesPackagePass{orcidlink}[2020/11/21]

\renewcommand\orcidlink[1]{%
    \texorpdfstring%
        {%
            \href%
                {https://orcid.org/#1}%
```

```
                {%
                    \begin{lateximage}[orcid #1]%   lwarp
                    \mbox{%
                        \scalerel*{%
                            \begin{tikzpicture}[yscale=-1,transform shape]
                            \pic{orcidlogo};
                            \end{tikzpicture}
                        }{|}%
                    }%
                    \end{lateximage}%   lwarp
                }%
        }%
        {}%
}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\orcidlink}[1]{}}
\end{warpMathJax}
```

File 351 **lwarp-overpic.sty**

§ 460 Package

overpic

(*Emulates or patches code by* Rolf Niepraschk.)

overpic (*Pkg*) overpic is patched for use by lwarp.

⚠ scaling The macros \overpicfontsize and \overpicfontskip are used during HTML generation. These are sent to \fontsize to adjust the font size for scaling differences between the print and HTML versions of the document. Renew these macros before using the overpic and Overpic environments.

See section 88.2 for the print-mode version of \overpicfontsize and \overpicfontskip.

for HTML output:

```
\LWR@ProvidesPackagePass{overpic}[2017/10/06]

\newcommand*{\overpicfontsize}{12}
\newcommand*{\overpicfontskip}{14}

\BeforeBeginEnvironment{overpic}{%
    \begin{lateximage}%
    \fontsize{\overpicfontsize}{\overpicfontskip}%
    \selectfont%
}

\AfterEndEnvironment{overpic}{\end{lateximage}}

\BeforeBeginEnvironment{Overpic}{%
    \begin{lateximage}%
    \fontsize{\overpicfontsize}{\overpicfontskip}%
    \selectfont%
}

\AfterEndEnvironment{Overpic}{\end{lateximage}}
```

File 352 **lwarp-pagegrid.sty**

§ 461 Package pagegrid

pagegrid (*Pkg*) pagegrid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pagegrid}[2016/05/16]
\newcommand*{\pagegridsetup}[1]{}
```

File 353 **lwarp-pagenote.sty**

§ 462 Package pagenote

pagenote (*Pkg*) pagenote works as-is, but the page option is disabled.

⚠ labels Note that labels in page notes do not appear as expected, even in the print version.

for HTML output:

```
\DeclareOption{page}{}
\LWR@ProvidesPackagePass{pagenote}[2009/09/03]
```

For MathJax:

```
\begin{warpMathJax}
\appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRpagenote}{\thepagenote}}
\CustomizeMathJax{\def\LWRpagenote{1}}
\CustomizeMathJax{\newcommand{\pagenote}[2][\LWRpagenote]{{}^{\mathrm{#1}}}}
\end{warpMathJax}
```

There is no \pagenotemark, so the following are not required:

```
\providecommand{\pagenotename}{pagenote}
\appto\LWR@syncnotenames{\LWR@synconenotename{LWRpagenote}{\pagenotename}}
```

File 354 **lwarp-pagesel.sty**

§ 463 Package pagesel

pagesel (*Pkg*) pagesel is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pagesel}[2016/05/16]
```

File 355 **lwarp-paralist.sty**

§ 464 Package paralist

(*Emulates or patches code by* Bernd Schandl.)

paralist (*Pkg*) paralist is supported with minor changes.

for HTML output:

```
\LWR@ProvidesPackagePass{paralist}[2017/01/22]
```

The compact environments are identical to the regular ones:

```
\LetLtxMacro\compactitem\itemize
\LetLtxMacro\compactenum\enumerate
\LetLtxMacro\compactdesc\description
\LetLtxMacro\endcompactitem\enditemize
\LetLtxMacro\endcompactenum\endenumerate
\LetLtxMacro\endcompactdesc\enddescription
```

For the inline environments, revert `\item` to its original print-mode version:

```
\AtBeginEnvironment{inparaitem}{\LetLtxMacro\item\LWR@origitem}
\AtBeginEnvironment{inparaenum}{\LetLtxMacro\item\LWR@origitem}
\AtBeginEnvironment{inparadesc}{\LetLtxMacro\item\LWR@origitem}
```

Manual formatting of the description labels:

```
\def\paradescriptionlabel#1{{\normalfont\textbf{#1}}}
```

File 356 **lwarp-parallel.sty**

§ 465 Package

parallel

(*Emulates or patches code by* Matthias Eckermann.)

parallel (*Pkg*) parallel is emulated.

Package options are ignored. Footnotes are treated as normal lwarp footnotes.

Environment option c gives side-by-side s of class minipage, each of whose width is a percent depending on the given left and right widths, proportional to `\linewidth`.

Inside each environment, `\linewidth` and `\textwidth` are set for the print-output sizes.

for HTML output: Discard all options for lwarp-parallel:

```
\LWR@ProvidesPackageDrop{parallel}[2003/04/13]
```

```
\newcounter{LWR@parallel@Lwidth}
\newcounter{LWR@parallel@Rwidth}
\newcommand*{\LWR@parallel@border}
\newenvironment*{Parallel}[3][]%
    {%
        \LWR@printpendingfootnotes%
        \setlength{\linewidth}{\LWR@userstextwidth}%
        \setlength{\textwidth}{\LWR@userstextwidth}%
        \renewcommand*{\LWR@parallel@border}{}%
        \ifstrequal{#1}{v}%
            {%
          \renewcommand*{\LWR@parallel@border}{ ; border-left: 2px solid black}%
```

```
            }%
            {}%
        \ifblank{#2}{%
            \ifblank{#3}{% {}{}
                \setcounter{LWR@parallel@Lwidth}{45}%
                \setcounter{LWR@parallel@Rwidth}{45}%
            }% {}{}
            {% {}{x}
                \setlength{\LWR@templengthone}{\linewidth-#3}%
                \setcounter{LWR@parallel@Lwidth}{%
                    90*\ratio{\LWR@templengthone}{\linewidth}%
                }%
                \setcounter{LWR@parallel@Rwidth}{%
                    90*\ratio{#3}{\linewidth}%
                }%
            }% {}{x}
        }% #2 blank
        {% #2 non-blank
            \ifblank{#3}{% {x}{}
                \setcounter{LWR@parallel@Lwidth}{%
                    90*\ratio{#2}{\linewidth}%
                }%
                \setlength{\LWR@templengthone}{\linewidth-#2}%
                \setcounter{LWR@parallel@Rwidth}{%
                    90*\ratio{\LWR@templengthone}{\linewidth}%
                }%
            }% {x}{}
            {% {x}{x}
                \setcounter{LWR@parallel@Lwidth}{%
                    90*\ratio{#2}{\linewidth}%
                }%
                \setcounter{LWR@parallel@Rwidth}{%
                    90*\ratio{#3}{\linewidth}%
                }%
            }% {x}{x}
        }% #2 non-blank
    }
    {%
        \ParallelAtEnd%
        \renewcommand*{\ParallelAtEnd}{}%
        \LWR@printpendingfootnotes%
    }

\newcommand*{\ParallelLText}[1]{%
    \begin{BlockClass}[%
        width:\arabic{LWR@parallel@Lwidth}\% ; % space
        padding: .5ex 1\% ; % space
    ]{minipage}%
    #1%
    \end{BlockClass}%
}

\newcommand*{\ParallelRText}[1]{%
    \begin{BlockClass}[%
        width:\arabic{LWR@parallel@Rwidth}\% ; % space
        padding: .5ex 1\% ; % space
        \LWR@parallel@border%
    ]{minipage}%
    #1%
    \end{BlockClass}%
```

```
}

\newcommand*{\ParallelPar}{\LWR@printpendingfootnotes}

\newcommand*{\ParallelAtEnd}{}
```

File 357 **lwarp-parcolumns.sty**

§ 466 Package

parcolumns

(*Emulates or patches code by* Jonathan Sauer.)

parcolumns (*Pkg*) parcolumns is emulated.

rulebetween is honored. The other keys are ignored, including colwidths.

Each column is placed inside a of class minipage, each of whose width is fixed at 85% divided by the number of columns. In most cases, this results in side-by-side minipages adapting to the browser width. Inside each minipage, \linewidth, \textwidth, and \textheight are set for a virtual 6×9 inch page, with \linewidth divided by the number of columns.

for HTML output: Discard all options for lwarp-parcolumns:

```
\RequirePackage{keyval}%

\LWR@ProvidesPackageDrop{parcolumns}[2004/11/25]

\newcounter{LWR@parcolumns@numcols}
\newcounter{LWR@parcolumns@thiscol}
\newcounter{LWR@parcolumns@width}
\newbool{LWR@parcolumns@started}
\newbool{LWR@parcolumns@rule}

\define@key{LWRparcols}{colwidths}{}
\define@key{LWRparcols}{distance}{}
\define@key{LWRparcols}{rulebetween}[true]{%
    \setbool{LWR@parcolumns@rule}{#1}%
}
\define@key{LWRparcols}{nofirstindent}{}
\define@key{LWRparcols}{sloppy}{}
\define@key{LWRparcols}{sloppyspaces}{}

\newenvironment*{parcolumns}[2][]
    {%
        \begin{LWR@setvirtualpage}*[#2]%
        \setcounter{LWR@parcolumns@numcols}{#2}%
        \setcounter{LWR@parcolumns@thiscol}{1}%
        \boolfalse{LWR@parcolumns@started}%
        \boolfalse{LWR@parcolumns@rule}%
        \setcounter{LWR@parcolumns@width}{%
            85/#2
        }%
        \setkeys{LWRparcols}{#1}%
    }
    {%
```

```
        \colplacechunks%
        \end{LWR@setvirtualpage}%
    }

\newcommand{\LWR@parcolumns@onecol}[1]{%
    \ifbool{LWR@parcolumns@started}%
        {}%
        {%
            \LWR@htmldivclass{parcolumns}%
            \booltrue{LWR@parcolumns@started}%
        }%
    \ifboolexpr{%
        bool {LWR@parcolumns@rule} and
        test {%
            \ifnumgreater
                {\value{LWR@parcolumns@thiscol}}
                {1}
        }%
    }%
        {\renewcommand{\LWR@tempone}{ ; border-left: 2px solid black}}%
        {\renewcommand{\LWR@tempone}{}}%
    \begin{BlockClass}[%
        width:\arabic{LWR@parcolumns@width}\% ; % space
        padding: .5ex 1\% ; % space
        \LWR@tempone%
    ]{minipage}%
    #1%
    \end{BlockClass}%
    \addtocounter{LWR@parcolumns@thiscol}{1}%
}

\newcommand{\colchunk}[2][\value{LWR@parcolumns@thiscol}]{%
    \whileboolexpr{%
        test {%
            \ifnumcomp%
                {\value{LWR@parcolumns@thiscol}}
                {<}
                {#1}%
        }%
    }{%
        \LWR@parcolumns@onecol{}%
    }%
    \LWR@parcolumns@onecol{#2}%
}

\newcommand*{\colplacechunks}{%
    \ifbool{LWR@parcolumns@started}%
        {%
            \LWR@htmldivclassend{div}%
            \boolfalse{LWR@parcolumns@started}%
        }%
        {}%
    \setcounter{LWR@parcolumns@thiscol}{1}%
}
```

File 358 **lwarp-parnotes.sty**

§ 467 Package

parnotes

(*Emulates or patches code by* Chelsea Hughes.)

parnotes (*Pkg*) parnotes is supported with some patches.

for HTML output:

```
\LWR@ProvidesPackagePass{parnotes}[2019/07/23]

\long\def\PN@parnote@real#1#2{%
    \parnotemark{#1}%
    % Unless this is the first parnote in \PN@text, add a separator first
    \unless\ifx\PN@text\@empty\g@addto@macro\PN@text{\parnoteintercmd}\fi
    % Redefine \@currentlabel to the parnote label, so \label works
    \g@addto@macro\PN@text{%
%         \phantomsection%
        \def\@currentlabel{#1}%
        \def\cref@currentlabel{%        lwarp
            [parnotemark][\arabic{parnotemark}][]\theparnotemark%
        }%
    }%
    \g@addto@macro\PN@text{%
        \LWR@textcurrentfont{%          lwarp
            \parnotemark{#1}\nolinebreak\thinspace#2%
        }%
    }%
}

\def\PN@parnotes@real{%

\ifPN@inparnotes
\else

    \LWR@stoppars%
```

Avoid nested paragraphs:

```
    \addtocounter{LWR@spandepth}{1}%

    % We call \par later, so this avoids recursion with \PN@parnotes@auto
    \PN@inparnotestrue
%     \unless\ifvmode\par\fi
    % Avoid page breaks between a paragraph and its parnotes
%     \nopagebreak\addvspace{\parnotevskip}%
    \begin{BlockClass}(note){footnotes}%    lwarp
    \leavevmode\LWR@orignewline%
```

Typeset the parnote inside its own group to avoid global changes:

```
    {%
        \parnotefmt{\PN@text}%
    }%
    \leavevmode\LWR@orignewline%
```

```
    \end{BlockClass}%                       lwarp

    \leavevmode\LWR@orignewline%
    \global\def\PN@text{}%
    %
    % These can be enabled or disabled by package options
    %
    \PN@disable@indent
    \PN@reset@optional
    \PN@inparnotesfalse
```

Reenable normal paragraph handling:

```
    \addtocounter{LWR@spandepth}{-1}%
\fi
}

\newbool{LWR@parnotes@doingauto}
\boolfalse{LWR@parnotes@doingauto}

\def\PN@parnotes@auto{%
    \ifbool{LWR@parnotes@doingauto}{
        \ifx\@currenvir\@PN@autopn
            \unless\ifPN@inparnotes
                \unless\ifx\PN@text\@empty
                    \expandafter\PN@parnotes@real
                \fi
            \fi
        \fi
    }{}%
}
```

Replace original logic due to the use of new LaTeXparagraph hook handling:

```
\renewenvironment{autopn}%
    {\booltrue{LWR@parnotes@doingauto}}
    {\PN@parnotes@auto}%
```

If cleveref is in use, name the new notes:

```
\AtBeginDocument{
    \ifdef{\crefname}{
        \crefname{parnotemark}{paragraph note}{paragraph notes}
        \Crefname{parnotemark}{Paragraph note}{Paragraph notes}
    }{}
}
```

To nullify the footnotes where necessary:

```
\apptocmd{\LWR@nullifyfootnotes}{%
    \renewcommand{\parnote}[2][]{}%
    \renewcommand\parnotemark[1]{}%
}{}{}
```

For MathJax:

```
\begin{warpMathJax}
```

```
\providecommand{\parnotename}{parnote}
\appto\LWR@syncnotenumbers{%
    \addtocounter{parnotemark}{-1}% specific to parnotes
    \LWR@synconenotenumber{LWRparnote}{\theparnotemark}%
    \addtocounter{parnotemark}{1}% specific to parnotes
}
\appto\LWR@syncnotenames{\LWR@synconenotename{LWRparnote}{\parnotename}}
\CustomizeMathJax{\def\LWRparnote{1}}
\CustomizeMathJax{\newcommand{\parnote}[2][\LWRparnote]{{}^{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\parnotemark}[1][\LWRparnote]{{}^{\mathrm{#1}}}}
\end{warpMathJax}
```

File 359 **lwarp-parskip.sty**

§ 468 Package parskip

parskip (*Pkg*) parskip is ignored.

for HTML output: Discard all options for lwarp-parskip.

```
\LWR@ProvidesPackageDrop{parskip}[2001/04/09]
```

File 360 **lwarp-pbalance.sty**

§ 469 Package pbalance

pbalance (*Pkg*) pbalance is ignored.

for HTML output:

```
\RequirePackage{balance}

\LWR@ProvidesPackageDrop{pbalance}[2022/07/28]

\newcommand\shrinkLastPage[1]{}
\newcommand\balancePageNum[1]{}
\newcommand\nopbalance{}
```

File 361 **lwarp-pbox.sty**

§ 470 Package pbox

(*Emulates or patches code by* Simon Law.)

pbox (*Pkg*) pbox is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{pbox}[2011/12/07]

\NewDocumentCommand{\pbox}{O{t} O{} O{t} m +m}{%
\global\booltrue{LWR@minipagefullwidth}%
\parbox[#1][#2][#3]{#4}{#5}%
}

```

```
\newcommand{\settominwidth}[3][\columnwidth]{%
\settowidth{#2}{#3}%
}

\newcommand{\widthofpbox}[1]{%
\widthof{#1}%
}
```

File 362 **lwarp-pdfcol.sty**

§ 471 Package pdfcol

pdfcol (*Pkg*) pdfcol is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcol}[2018/11/01]

\ltx@newif\ifpdfcolAvailable
\pdfcolAvailablefalse

\def\pdfcolErrorNoStacks{
    \PackageInfo{lwarp-pdfcol}{Ignoring pdfcol for HTML output.}
}

\def\pdfcolInitStack#1{}%

\long\def\pdfcolIfStackExists#1#2#3{#3}%

\def\pdfcolSwitchStack#1{}%

\def\pdfcolSetCurrentColor{}%

\def\pdfcolSetCurrent#1{}%
```

File 363 **lwarp-pdfcolfoot.sty**

§ 472 Package pdfcolfoot

pdfcolfoot (*Pkg*) pdfcolfoot is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcolfoot}[2016/05/16]

\newcommand*{\pdfcolfoot@switch}{}

\newcommand*{\pdfcolfoot@current}{}
```

File 364 **lwarp-pdfcolmk.sty**

§ 473 Package pdfcolmk

pdfcolmk (*Pkg*) pdfcolmk is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcolmk}[2016/05/16]
```

File 365 **lwarp-pdfcolparallel.sty**

§ 474 Package

pdfcolparallel

pdfcolparallel (*Pkg*) pdfcolparallel is ignored.

for HTML output:

```
\RequirePackage{keyval}%

\LWR@ProvidesPackageDropA{pdfcolparallel}{2016/05/16}
```

Pass options to parallel:

```
\DeclareOption*{%
    \PassoptionsToPackage{\CurrentOption}{parallel}%
}
```

Process the options:

```
\LWR@ProvidesPackageDropB
```

Require parallel with the given options:

```
\RequirePackage{parallel}[2003/04/13]
```

Ignore the new key:

```
\define@key{parallel}{rulebetweencolor}{}
```

File 366 **lwarp-pdfcolparcolumns.sty**

§ 475 Package

pdfcolparcolumns

pdfcolparcolumns (*Pkg*) pdfcolparcolumns is ignored.

for HTML output:

```
\LWR@ProvidesPackageDropA{pdfcolparcolumns}{2016/05/16}
```

Pass options to parcolumns:

```
\DeclareOption*{%
    \PassoptionsToPackage{\CurrentOption}{parcolumns}%
}
```

Process the options:

```
\LWR@ProvidesPackageDropB
```

Require parcolumns with the given options:

```
\RequirePackage{parcolumns}[2004/11/25]
```

Ignore the new key:

```
\define@key{LWRparcols}{rulebetweencolor}{}
```

File 367 **lwarp-pdfcomment.sty**

§ 476 Package pdfcomment

pdfcomment (*Pkg*) pdfcomment is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcomment}[2016/06/13]

\newenvironment{pdfsidelinecomment}[2][]{}{}
\newcommand{\pdfcomment}[2][]{}
\newcommand{\pdfmargincomment}[2][]{}
\newcommand{\pdfmarkupcomment}[3][]{#2}
\newcommand{\pdffreetextcomment}[2][]{}
\newcommand{\pdfsquarecomment}[2][]{}
\newcommand{\pdfcirclecomment}[2][]{}
\newcommand{\pdflinecomment}[2][]{}
\newcommand{\pdftooltip}[3][]{#2}
\newcommand{\pdfcommentsetup}[2][]{}
\newcommand{\listofpdfcomments}[1][]{}
\newcommand{\setliststyle}[1]{}
\newcommand{\defineliststyle}[2]{}
\newcommand{\defineavatar}[2]{}
\newcommand{\definestyle}[2]{}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\pdfmarkupcomment}[3][]{#2}}
\CustomizeMathJax{\newcommand{\pdftooltip}[3][]{#2}}
\end{warpMathJax}
```

File 368 **lwarp-pdfcrypt.sty**

§ 477 Package pdfcrypt

pdfcrypt (*Pkg*) pdfcrypt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcrypt}[2016/05/16]

\newcommand*{\pdfcryptsetup}[1]{}
```

File 369 **lwarp-pdflscape.sty**

§ 478 Package pdflscape

pdflscape (*Pkg*) pdflscape is ignored.

for HTML output: Discard all options for lwarp-pdflscape:

```
\LWR@ProvidesPackageDrop{pdflscape}[2019/12/05]

\let\landscape\relax
\let\endlandscape\relax

\newenvironment*{landscape}{}{}
```

File 370 **lwarp-pdfmarginpar.sty**

§ 479 Package

pdfmarginpar

pdfmarginpar (*Pkg*) pdfmarginpar is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfmarginpar}[2011/08/05]

\newcommand{\pdfmarginpar}[2][]{}
\newcommand{\pdfmarginparset}[1]{}
```

File 371 **lwarp-pdfpages.sty**

§ 480 Package

pdfpages

(*Emulates or patches code by* Andreas Matthias.)

pdfpages (*Pkg*) pdfpages is patched for use by lwarp.

Option `link` and `linkname` work:

```
\hyperlink{<filename>.pdf.<pagenubmer}{some text}
\hyperlink{<linkname>.<pagenubmer}{some text}
```

Options which make no sense in HTML are disabled.

for HTML output:

```
\LWR@ProvidesPackagePass{pdfpages}[2022-10-09]
```

Disable option which have no meaning for HTML output:

```
\define@key{pdfpages}{fitpaper}[false]{}
\define@key{pdfpages}{landscape}[false]{}
\define@key{pdfpages}{openright}[false]{}
\define@key{pdfpages}{signature}{}
\define@key{pdfpages}{signature*}{}
\define@key{pdfpages}{booklet}[false]{}
\define@key{pdfpages}{rotateoversize}[false]{}
\define@key{pdfpages}{doublepages}[false]{}
\define@key{pdfpages}{doublepagestwist}[false]{}
\define@key{pdfpages}{doublepagestwistodd}[false]{}
\define@key{pdfpages}{doublepagestwist*}[false]{}
\define@key{pdfpages}{doublepagestwistodd*}[false]{}
```

```
\define@key{pdfpages}{duplicatepages}[2]{}
\define@key{pdfpages}{thread}[false]{}
\define@key{pdfpages}{threadname}{}
\define@key{pdfpages}{linkfit}{}
\define@key{pdfpages}{linktodoc}[false]{}
\define@key{pdfpages}{linktodocfit}{}
\define@key{pdfpages}{linkfilename}{}
\define@key{pdfpages}{survey}[false]{}
\define@key{pdfpages}{survey-nolink}[false]{}
\define@key{pdfpages}{newwindow}[false]{}
```

Use print mode while measuring the page numbers:

```
\xpretocmd{\AM@getpagecount}{\LWR@restoreorigformatting}{}{}
```

Emulate a bit of eso-pic:

```
\newif\ifESO@texcoord

\newcommand{\ESO@HookIIBG}{}

\renewcommand{\AM@AddToShipoutPicture}{\g@addto@macro\ESO@HookIIBG}

\renewcommand{\ClearShipoutPicture}{}
```

`\LWR@esopic@newpage` At each `\newpage`.

```
\newcommand*{\LWR@esopic@newpage}{%
```

Is there something to draw?

```
\ifdefvoid{\ESO@HookIIBG}%
{}%
{%
```

If the `link` option was specified, add a hyper target:

```
    \ifAM@link%
        \hypertarget{\AM@linkname.\AM@page}{}%
    \fi%
```

Draw inside a picture environment of the size of a virtual page:

```
    \begingroup%
    \setlength{\unitlength}{1in}%
    \begin{picture}(8,10.5)%
    \ESO@HookIIBG%
    \end{picture}%
    \endgroup%
    \global\let\ESO@HookIIBG\@empty%
}
}
```

`\AM@output` Patched to use `\LWR@esopic@newpage`.

```
\xpatchcmd{\AM@output@i}
    {\newpage}
    {\LWR@esopic@newpage}
    {}
    {\LWR@patcherror{pdfpages}{AM@output-1}}

```

```
\xpatchcmd{\AM@output@i}
    {\newpage}
    {\LWR@esopic@newpage}
    {}
    {\LWR@patcherror{pdfpages}{AM@output-2}}

\xpatchcmd{\AM@output@i}
    {\newpage}
    {\LWR@esopic@newpage}
    {}
    {\LWR@patcherror{pdfpages}{AM@output-3}}
```

`\includepdf` Patched to set the user's paper size.

```
\xpretocmd{\includepdf}{%
    \begingroup%
    \setlength{\paperwidth}{\LWR@userspaperwidth}%
    \setlength{\paperheight}{\LWR@userspaperheight}%
}{}{}

\xapptocmd{\includepdf}{%
    \endgroup%
}{}{}
```

`\includepdfmerge` Patched to set the user's paper size.

```
\xpretocmd{\includepdfmerge}{%
    \begingroup%
    \setlength{\paperwidth}{\LWR@userspaperwidth}%
    \setlength{\paperheight}{\LWR@userspaperheight}%
}{}{}

\xapptocmd{\includepdfmerge}{%
    \endgroup%
}{}{}
```

`\AM@hyper@begin@i` Hyper links are created by `\LWR@esopic@newpage`, so don't create them here:

```
\renewcommand{\AM@hyper@begin@i}{}
```

File 372 **lwarp-pdfprivacy.sty**

§ 481 Package pdfprivacy

`pdfprivacy` (*Pkg*) pdfprivacy is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfprivacy}[2017/12/03]
```

File 373 **lwarp-pdfrender.sty**

§ 482 Package pdfrender

`pdfrender` (*Pkg*) pdfrender is allowed during HTML, but it has no effect on HTML text output. pdfrender is enabled for use with xfakebold, and it is enabled during HTML so that

it may be in use when an SVG math image is started. I.e. xfakebold's `\setBold` may be used outside of a math expression and still be detected when the math begins.

The lwarp-pdfrender package is present because it used to disable pdfrender, so this newer version is to overwrite older versions.

for HTML output:

```
\LWR@ProvidesPackagePass{pdfrender}[2019/12/29]
```

File 374 **lwarp-pdfsync.sty**

§ 483 Package

pdfsync

(*Emulates or patches code by* J. Laurens.)

`pdfsync` (*Pkg*) pdfsync is ignored.

for HTML output: Discard all options for lwarp-pdfsync:

```
\LWR@ProvidesPackageDrop{pdfsync}[2008/01/26]

\newcommand*{\pdfsync}{}
\newcommand*{\pdfsyncstart}{}
\newcommand*{\pdfsyncstop}{}
```

File 375 **lwarp-pdftricks.sty**

§ 484 Package

pdftricks

(*Emulates or patches code by* C. V. Radhakrishnan, C. V. Rajagopal, Antoine Chambert-Loir.)

`pdftricks` (*Pkg*) pdftricks is patched for use by lwarp.

⚠ convert image files The pdftricks image files `<jobname>-fig*.pdf` must be converted to `.svg`, or else a missing file error will occur. The image files must also be converted again whenever they change. To convert the images:

Enter ⇒ **`lwarpmk pdftosvg <jobname>-fig*.pdf`**

for HTML output:

```
\LWR@ProvidesPackagePass{pdftricks}[2003/08/10]
```

Reuse the print-mode images:

```
\def\PDFTfigname{\BaseJobname-fig\thepsfig}
```

If the `.pdf` images have not yet been converted to `.svg` then an error about a missing file will occur. Warn the user to convert the images.

```
\PackageWarning{lwarp-pdftricks}{%
When the pdftricks images change,
remember to convert PDF images to SVG using 'lwarpmk pdftosvg *-fig.pdf',
}

\AfterEndDocument{\typeout{***}}
```

```
\AfterEndDocument{\typeout{*** Note: If pdftricks images are not found, new, or updated,}}
\AfterEndDocument{\typeout{*** \space use 'lwarpmk pdftosvg \BaseJobname-fig*.pdf'}}
\AfterEndDocument{\typeout{***}}
```

File 376 **lwarp-pdfx.sty**

§ 485 Package pdfx

pdfx (*Pkg*) pdfx is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfx}[2017/05/18]
```

File 377 **lwarp-perpage.sty**

§ 486 Package perpage

(*Emulates or patches code by* David Kastrup.)

perpage (*Pkg*) perpage is mostly ignored, but support is added for footnote counters.

There is no page number in HTML, so most counters are not reset. If the document redefines `\the<countername>` to include `\theperpage`, it is necessary to place that redefininition inside a `warpprint` environment to avoid modifying the HTML defintions.

`\AddAbsoluteCounter` must not be inside `warpprint`, as the counter must be added for HTML also, although it is not incremented.

footnote numbering To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
— or —
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by `FootnoteDepth`, which is not necessarily by HTML page. This is recommended for `\alph`, `\Alph`, or `\fnsymbol` footnotes, due to the limited number of symbols which are available.

for HTML output:

```
\LWR@ProvidesPackageDrop{perpage}[2014/10/25]

\newcommand\AddAbsoluteCounter[1]
{
    \@ifundefined{c@abs#1}{%
        \expandafter\newcount\csname c@abs#1\endcsname
        \global\value{abs#1}\@ne
%         \global\expandafter\let\csname cl@abs#1\endcsname\@empty
        \expandafter\xdef\csname theabs#1\endcsname{%
%             \noexpand\number \csname c@abs#1\endcsname
```

```
        }%
%         \global\@namedef{c@pabs@#1}{\pp@cl@begin
%         \stepcounter{abs#1}%
%         \pp@cl@end}%
%         \@addtoreset{pabs@#1}{#1}
    }
    {}
}

\AddAbsoluteCounter{page}
\def\theabspage{1}

\newcommand*\MakePerPage[2][1]{%
    \ifltxcounter{#2Reset}{%
        \setcounter{#2Reset}{#1}%
    }{

}%
}

\newcommand*\MakeSorted[1]{}

\newcommand*\MakeSortedPerPage[2][1]{%
    \ifltxcounter{#2Reset}{%
        \setcounter{#2Reset}{#1}%
    }{
}%
}

\newcommand*{\theperpage}{1}
```

File 378 **lwarp-pfnote.sty**

§ 487 Package pfnote

pfnote (*Pkg*) pfnote is ignored.

pfnote

⚠ pfnote numbers

While emulating pfnote, lwarp is not able to reset HTML footnote numbers per page number to match the printed version, as HTML has no concept of page numbers. lwarp therefore uses continuous footnote numbering even for pfnote.

for HTML output:

```
\LWR@ProvidesPackageDrop{pfnote}[1999/07/14]
```

File 379 **lwarp-phfqit.sty**

§ 488 Package phfqit

(*Emulates or patches code by* Philippe Faist.)

phfqit (*Pkg*) phfqit is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{phfqit}[2017/08/16]

\LetLtxMacro\LWR@origbitstring\bitstring
```

```
3
4 \renewcommand\bitstring[1]{%
5 \InlineClass[%
6     text-decoration: overline underline ;
7 ]{bitstring}{#1}%
8 % \phfqit@bitstring{#1}%
9 }
10
11 \appto\LWR@restoreorigformatting{%
12 \LetLtxMacro\bitstring\LWR@origbitstring%
13 }
```

File 380 **lwarp-physics.sty**

§ 489 Package physics

(*Emulates or patches code by* Sergio C. de la Barrera.)

physics (*Pkg*) physics works as-is for HTML with SVG math.

For MathJax, the MathJax v3 physics extension is used.

for HTML output:

```
1 \LWR@ProvidesPackagePass{physics}% No date is provided by the package.

2 \begin{warpMathJax}
3 \PackageNoteNoLine{lwarp, physics}{The MathJax v3 extension will be used}
4 \CustomizeMathJax{\require{physics}}
5 \end{warpMathJax}
```

File 381 **lwarp-physunits.sty**

§ 490 Package physunits

(*Emulates or patches code by* Brian W. Mulligan.)

physunits (*Pkg*) physunits is supported as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
1 \LWR@ProvidesPackagePass{physunits}[2020/03/26]

2 \begin{warpMathJax}
3 \LWR@infoprocessingmathjax{physunits}
4
5 \CustomizeMathJax{\newcommand{\micro}{\mu}}
6 \CustomizeMathJax{\newcommand{\V}[1][ ]{\,\mathrm{#1V}}}
7 \CustomizeMathJax{\newcommand{\Volt}[1][ ]{\,\mathrm{#1V}}}
8 \CustomizeMathJax{\newcommand{\Coulomb}[1][ ]{\,\mathrm{#1C}}}
9 \CustomizeMathJax{\newcommand{\esu}{\,\mathrm{esu}}}
10 \CustomizeMathJax{\newcommand{\Ohm}[1][ ]{\,\mathrm{#1\Omega}}}
11 \CustomizeMathJax{\newcommand{\Amp}[1][ ]{\,\mathrm{#1A}}}
12 \CustomizeMathJax{\newcommand{\Farad}[1][ ]{\,\mathrm{#1F}}}
13 \CustomizeMathJax{\newcommand{\Tesla}[1][ ]{\,\mathrm{#1T}}}
14 \CustomizeMathJax{\newcommand{\Gauss}[1][ ]{\,\mathrm{#1G}}}
15 \CustomizeMathJax{\newcommand{\Henry}[1][ ]{\,\mathrm{#1H}}}
16 \CustomizeMathJax{\newcommand{\eV}[1][ ]{\,\mathrm{#1eV}}}
```

```
\CustomizeMathJax{\newcommand{\keV}{\,\mathrm{keV}}}
\CustomizeMathJax{\newcommand{\MeV}{\,\mathrm{MeV}}}
\CustomizeMathJax{\newcommand{\J}[1][ ]{\,\mathrm{#1J}}}
\CustomizeMathJax{\newcommand{\Joule}[1][ ]{\,\mathrm{#1J}}}
\CustomizeMathJax{\newcommand{\erg}{\,\mathrm{erg}}}
\CustomizeMathJax{\newcommand{\kcal}{\,\mathrm{kcal}}}
\CustomizeMathJax{\newcommand{\Cal}{\,\mathrm{Cal}}}
\CustomizeMathJax{\newcommand{\calorie}[1][ ]{\,\mathrm{#1cal}}}
\CustomizeMathJax{\newcommand{\BTU}{\,\mathrm{BTU}}}
\CustomizeMathJax{\newcommand{\tnt}{\,\mathrm{ton\, of\, TNT}}}
\CustomizeMathJax{\newcommand{\Watt}[1][ ]{\,\mathrm{#1W}}}
\CustomizeMathJax{\newcommand{\hpi}{\,\mathrm{hp(I)}}}
\CustomizeMathJax{\newcommand{\hpm}{\,\mathrm{hp(M)}}}
\CustomizeMathJax{\newcommand{\hp}{\,\mathrm{hp}}}
\CustomizeMathJax{\newcommand{\meter}[1][ ]{\,\mathrm{#1m}}}
\CustomizeMathJax{\newcommand{\m}[1][ ]{\,\mathrm{#1m}}}
\CustomizeMathJax{\newcommand{\km}{\,\mathrm{km}}}
\CustomizeMathJax{\newcommand{\au}{\,\mathrm{au}}}
\CustomizeMathJax{\newcommand{\pc}[1][ ]{\,\mathrm{#1pc}}}
\CustomizeMathJax{\newcommand{\ly}[1][ ]{\,\mathrm{#1ly}}}
\CustomizeMathJax{\newcommand{\cm}{\,\mathrm{cm}}}
\CustomizeMathJax{\newcommand{\nm}{\,\mathrm{nm}}}
\CustomizeMathJax{\newcommand{\ft}{\,\mathrm{ft}}}
\CustomizeMathJax{\newcommand{\inch}{\,\mathrm{in}}}
\CustomizeMathJax{\newcommand{\mi}{\,\mathrm{mi}}}
\CustomizeMathJax{\newcommand{\s}[1][ ]{\,\mathrm{#1s}}}
\CustomizeMathJax{\newcommand{\Sec}[1][ ]{\,\mathrm{#1s}}}
\CustomizeMathJax{\newcommand{\Min}{\,\mathrm{min}}}
\CustomizeMathJax{\newcommand{\h}{\,\mathrm{h}}}
\CustomizeMathJax{\newcommand{\y}[1][ ]{\,\mathrm{#1y}}}
\CustomizeMathJax{\newcommand{\Day}{\,\mathrm{d}}}

\CustomizeMathJax{\newcommand{\gm}[1][ ]{\,\mathrm{#1g}}}
\CustomizeMathJax{\newcommand{\kg}{\,\mathrm{kg}}}
\CustomizeMathJax{\newcommand{\lb}{\,\mathrm{lb}}}
\CustomizeMathJax{\newcommand{\amu}{\,\mathrm{amu}}}
\CustomizeMathJax{\newcommand{\N}[1][ ]{\,\mathrm{#1N}}}
\CustomizeMathJax{\newcommand{\Newton}[1][ ]{\,\mathrm{#1N}}}
\CustomizeMathJax{\newcommand{\dyne}[1][ ]{\,\mathrm{#1dyn}}}
\CustomizeMathJax{\newcommand{\lbf}{\,\mathrm{lbf}}}
\CustomizeMathJax{\newcommand{\kmps}{\,\mathrm{km}\,\mathrm{s}^{-1}}}
\CustomizeMathJax{\newcommand{\kmph}{\,\mathrm{km}\,\mathrm{h}^{-1}}}
\CustomizeMathJax{\newcommand{\mps}[1][ ]{\,\mathrm{#1m}\,\mathrm{s}^{-1}}}
\CustomizeMathJax{\newcommand{\miph}{\,\mathrm{mi}\,\mathrm{h}^{-1}}}
\CustomizeMathJax{\newcommand{\kts}{\,\mathrm{kts}}}

\CustomizeMathJax{\newcommand{\mpss}[1][ ]{\,\mathrm{#1m}\,\mathrm{s}^{-2}}}
\CustomizeMathJax{\newcommand{\gacc}{\,\mathrm{g}}}
\CustomizeMathJax{\newcommand{\ftpss}{\,\mathrm{ft}\,\mathrm{s}^{-2}}}
\CustomizeMathJax{\newcommand{\K}[1][ ]{\,\mathrm{#1K}}}
\CustomizeMathJax{\newcommand{\Kelvin}[1][ ]{\,\mathrm{#1K}}}
\CustomizeMathJax{\newcommand{\Celcius}{\,^\circ{\mathrm{C}}}}
\CustomizeMathJax{\newcommand{\Rankine}{\,^\circ{\mathrm{R}}}}
\CustomizeMathJax{\newcommand{\Fahrenheit}{\,^\circ{\mathrm{F}}}}

\CustomizeMathJax{\newcommand{\rpm}{\,\mathrm{rev}\,\Min^{-1}}}

\CustomizeMathJax{\newcommand{\Hz}[1][ ]{\,\mathrm{#1Hz}}}
\CustomizeMathJax{\newcommand{\barP}[1][ ]{\,\mathrm{#1bar}}}
\CustomizeMathJax{\newcommand{\atm}{\,\mathrm{atm}}}
```

```
\CustomizeMathJax{\newcommand{\Pa}[1][ ]{\,\mathrm{#1Pa}}}
\CustomizeMathJax{\newcommand{\mmHg}{\,\mathrm{mmHg}}}
\CustomizeMathJax{\newcommand{\inHg}{\,\mathrm{inHg}}}
\CustomizeMathJax{\newcommand{\lbsi}{\,\mathrm{psi}}}
\CustomizeMathJax{\newcommand{\lbsf}{\,\mathrm{psf}}}
\CustomizeMathJax{\newcommand{\Ba}[1][ ]{\,\mathrm{#1Ba}}}
\CustomizeMathJax{\newcommand{\Torr}[1][ ]{\,\mathrm{#1Torr}}}
\CustomizeMathJax{\newcommand{\mol}{\,\mathrm{mol}}}
\end{warpMathJax}
```

File 382 **lwarp-picinpar.sty**

§ 491 Package picinpar

(*Emulates or patches code by* Friedhelm Sowa.)

picinpar (*Pkg*) picinpar is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{picinpar}% No date is assigned.
```

The window is floated by a BlockClass style.

```
\long\def\LWR@HTML@window[#1,#2,#3,#4] {%
    \if #2r%
        \begin{BlockClass}[float:right](note){marginblock}%
    \else%
        \begin{BlockClass}[float:left](note){marginblock}%
    \fi%
    #3\par%
    #4%
    \end{BlockClass}%
}

\def\endLWR@HTML@window{}

\LWR@formattedenv{window}
```

The framepic and wframepic are placed inside a BlockClass of class framebox.

```
\def\LWR@HTML@framepic#1{%
    \begin{BlockClass}{framebox}
    \expandafter\box\csname #1box\endcsname%
    \end{BlockClass}
}
\LWR@formatted{framepic}

\def\LWR@HTML@wframepic#1{%
    \begin{BlockClass}{framebox}
    \expandafter\box\csname #1box\endcsname%
    \end{BlockClass}
}
\LWR@formatted{wframepic}
```

The caption is placed inside a BlockClass of class figurecaption.

```
\long\def\LWR@HTML@@makewincaption#1#2{%
```

```
\begin{BlockClass}{figurecaption}
#1: #2
\end{BlockClass}
}
\LWR@formatted{@makewincaption}
```

With HTML output, `figwindow` and `tabwindow` must not pre-decrement their counters.

```
\long\def\LWR@HTML@figwindow[#1,#2,#3,#4] {%
%       \advance\c@figure -1
      \window[#1,#2,{#3},{\def\@captype{figure}%
         \wincaption#4\par}] }

\def\endLWR@HTML@figwindow{\endwindow}

\LWR@formattedenv{figwindow}
```

For `tabwindow`, to change the catcode of `&`, `\StartDefiningTabulars` is used before absorbing the arguments, and `\EndDefiningTabulars` is used at the end of the environment.

```
\long\def\LWR@HTML@subtabwindow[#1,#2,#3,#4] {%
%       \advance\c@table -1
      \window[#1,#2,{#3},{\def\@captype{table}%
         \wincaption#4\par}] }

\newcommand*{\LWR@HTML@tabwindow}{%
    \StartDefiningTabulars%
    \LWR@HTML@subtabwindow%
}

\def\endLWR@HTML@tabwindow{%
    \endwindow%
    \StopDefiningTabulars%
}

\LWR@formattedenv{tabwindow}
```

File 383 **lwarp-pifont.sty**

§ 492 Package

pifont

(*Emulates or patches code by* Walter Schmidt.)

`pifont` (*Pkg*) pifont is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

for HTML output:

```
\LWR@ProvidesPackagePass{pifont}[2005/04/12]

\renewcommand{\Pisymbol}[2]{%
    \begin{lateximage}*[Pisymbol][pisymbol#1#2]%
    {\Pifont{#1}\char#2}%
    \end{lateximage}%
}

```

```
\newcommand{\LWR@HTML@Pifill}[2]{
    \Pisymbol{#1}{#2} \Pisymbol{#1}{#2} \Pisymbol{#1}{#2}
}
\LWR@formatted{Pifill}

\newcommand{\LWR@HTML@Piline}[2]{%
    \par\noindent\hspace*{0.5in}
    \Pifill{#1}{#2} \Pifill{#1}{#2} \Pifill{#1}{#2}
}
\LWR@formatted{Piline}
```

File 384 **lwarp-pinlabel.sty**

§ 493 Package pinlabel

(*Emulates or patches code by* Colin Rourke.)

pinlabel (*Pkg*) pinlabel is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{pinlabel}% no date given

\xpretocmd{\psfig}
    {\begin{leteximage}[-pinlabel-~\PackageDiagramAltText]}
    {}
    {\LWR@patcherror{pinlabel}{psfigA}}

\xapptocmd{\psfig}
    {\end{leteximage}}
    {}
    {\LWR@patcherror{pinlabel}{psfigB}}
```

File 385 **lwarp-placeins.sty**

§ 494 Package placeins

(*Emulates or patches code by* Donald Arseneau.)

placeins (*Pkg*) placeins is ignored.

Discard all options for lwarp-placeins:

for HTML output:

```
\LWR@ProvidesPackageDrop{placeins}[2005/04/18]

\newcommand*{\FloatBarrier}{}
```

File 386 **lwarp-plarydshln.sty**

§ 495 Package plarydshln

plarydshln (*Pkg*) plarydshln is emulated by lwarp-arydshln.

for HTML output:

```
\LWR@ProvidesPackageDrop{plarydshln}[2018/10/20]
```

```
\LWR@origRequirePackage{lwarp-arydshln}
```

File 387 **lwarp-plext.sty**

§ 496 Package

plext

plext (*Pkg*) plext is preloaded by jtarticle and related classes.

for HTML output:

```
\LWR@loadbefore{plext}

\LWR@ProvidesPackagePass{plext}[2017/07/21]

\let\tate\relax

\DeclareExpandableDocumentCommand{\rensuji}{s o m}{#3}

% \layoutfloat(width,height)[pos]#4
\DeclareDocumentCommand{\layoutfloat}{d() o m}{}

% \DeclareLayoutCaption{type} <dir>(width)[pos1pos2]
\DeclareDocumentCommand{\DeclareLayoutCaption}{m d<> d() o}{}

\LetLtxMacro\pcaption\caption

% \layoutcaption<dir>(width)[pos]
\DeclareDocumentCommand{\layoutcaption}{d<> d() o}{}

\let\captiondir\relax
```

Add the optional <t/y> direction:

```
\RenewDocumentEnvironment{LWR@HTML@minipage}{d<> O{t} O{} O{t} m}
    {\LWR@HTML@sub@minipage{#2}{#3}{#4}{#5}}
    {\endLWR@HTML@sub@minipage}

\RenewDocumentCommand{\LWR@HTML@parbox}{d<> O{t} O{} O{t} m +m}
{
\LWR@traceinfo{parbox of width #4}%
\begin{minipage}[#2][#3][#4]{#5}%
#6
\end{minipage}%
}

% \pbox <t/y> [width] [l/r] {contents}
\RenewDocumentCommand{\pbox}{d<> O{0pt} O{c} m}{%
\global\booltrue{LWR@minipagefullwidth}%
\parbox{#2}{#4}%
}
```

picture, as modified by pext, is encapsulated by the lwarp core.

File 388 **lwarp-plextarydshln.sty**

§ 497 Package

plextarydshln

plextarydshln (*Pkg*) plextarydshln is emulated by lwarp-arydshln.

for HTML output:

```
\LWR@ProvidesPackageDrop{plextarydshln}[2018/10/20]
\LWR@origRequirePackage{lwarp-arydshln}
```

File 389 **lwarp-plextcolortbl.sty**

§ 498 Package

plextcolortbl

plextcolortbl (*Pkg*) plextcolortbl is emulated by lwarp-colortbl.

for HTML output:

```
\LWR@ProvidesPackageDrop{plextcolortbl}[2018/09/19]
\LWR@origRequirePackage{lwarp-colortbl}
```

File 390 **lwarp-plimsoll.sty**

§ 499 Package

plimsoll

(*Emulates or patches code by* Palle Jørgensen.)

plimsoll (*Pkg*) plimsoll is used as-is for svg math, and emulated for MathJax.

The circ option is honored. For MathJax, \plimsollsans is the same as \plimsollroman.

for HTML output:

```
\LWR@ProvidesPackagePass{plimsoll}[2020/10/09]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\plimsollroman}{\unicode{x029B5}}}

\CustomizeMathJax{\let\plimsoll\plimsollroman}
\CustomizeMathJax{\let\plimsollsans\plimsoll}

\ifdefstring{\stst}{^{\circ}}
    {\CustomizeMathJax{\newcommand{\stst}{^{\circ}}}}
    {\CustomizeMathJax{\newcommand{\stst}{^{\plimsoll}}}}
\end{warpMathJax}
```

File 391 **lwarp-prelim2e.sty**

§ 500 Package

prelim2e

(*Emulates or patches code by* Martin Schröder.)

prelim2e (*Pkg*) prelim2e is ignored.

for HTML output: Discard all options for lwarp-prelim2e:

```
\LWR@ProvidesPackageDrop{prelim2e}[2009/05/29]

\newcommand{\PrelimText}{}
\newcommand{\PrelimTextStyle}{}
\newcommand{\PrelimWords}{}
```

File 392 **lwarp-prettyref.sty**

§ 501 Package prettyref

(*Emulates or patches code by* Kevin S. Ruland.)

prettyref (*Pkg*) prettyref is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{prettyref}[1998/07/09]

\newrefformat{fig}{Figure \ref{#1}}
\newrefformat{tab}{Table \ref{#1}}
```

File 393 **lwarp-preview.sty**

§ 502 Package preview

preview (*Pkg*) preview is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{preview}[2017/04/24]

\newenvironment{preview}{}{}
\newenvironment{nopreview}{}{}
\NewDocumentCommand{\PreviewMacro}{s o o +m}{}
\NewDocumentCommand{\PreviewEnvironment}{s o o +m}{}
\newcommand{\PreviewSnarfEnvironment}[2][]{}
\NewDocumentCommand{\PreviewOpen}{s o}{}
\NewDocumentCommand{\PreviewClose}{s o}{}
\let\ifPreview\iffalse% \fi for syntax highlighting
```

File 394 **lwarp-psfrag.sty**

§ 503 Package psfrag

(*Emulates or patches code by* Michael C. Grant, David Carlisle.)

psfrag (*Pkg*) psfrag is patched for use by lwarp.

⚠ use psfrags The psfrags environment is modified to use lateximage to encapsulate the image. Always use a psfrags environment to contain any local \psfrag macros and the

associated \includegraphics or \epsfig calls. Outside of a psfrags environment, psfrags adjustments will not be seen by lwarp.

⚠ Tip: Use a mono-spaced font for the tags in the EPS file.

for HTML output:

```
\LWR@ProvidesPackagePass{psfrag}[1998/04/11]
```

A lateximage captures the modified image from the document.

```
\BeforeBeginEnvironment{psfrags}{%
    \begin{lateximage}[-psfrags-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{psfrags}{\end{lateximage}}
```

File 395 **lwarp-psfragx.sty**

§ 504 Package psfragx

(*Emulates or patches code by* Pascal Kockaert.)

psfragx (*Pkg*) psfragx is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{psfragx}[2012/05/02]
```

A lateximage captures the modified image from the document.

```
\def\pfx@includegraphicx#1#2{%
    \begin{lateximage}[-psfragx-~\PackageDiagramAltText]%
    \mbox{\pfx@overpix{#1}{#2}\endpfx@overpix}%
    \end{lateximage}%
}

\def\@@@overpix[#1]<#2>[#3]#4{%
    \begin{lateximage}[-psfragx-~\PackageDiagramAltText]%
    \pfx@overpix{#1,ovpfgd={#2},ovpbgd={#3}}{#4}%
}

\def\endoverpix{%
    \endpfx@overpix%
    \end{lateximage}%
}
```

File 396 **lwarp-pst-eps.sty**

§ 505 Package pst-eps

(*Emulates or patches code by* Herbert Voss.)

pst-eps (*Pkg*) pst-eps is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{pst-eps}[2005/05/20]

\renewenvironment{TeXtoEPS}{}{}
\renewcommand{\PSTtoEPS}[3][]{}
```

File 397 **lwarp-pstool.sty**

§ 506 Package pstool

(*Emulates or patches code by* Zebb Prime, Will Robertson.)

pstool (*Pkg*) pstool is patched for use by lwarp.

\graphicspath is ignored, and the file directory must be stated.

⚠ path and filename The filename must not have a file extension.

Use

Enter ⇒ **lwarpmk html**

followed by

Enter ⇒ **lwarpmk limages**

.

for HTML output:

```
\LWR@ProvidesPackagePass{pstool}[2018/01/20]
```

Each image is placed inside a lateximage to capture the results of psfrag.

```
\renewcommand\pstool@alwaysprocess[3][]{%
    \begin{lateximage}[-pstool-~\PackageDiagramAltText]%
    \includegraphics{#2.pdf}%
    \end{lateximage}%
}
\LetLtxMacro\pstool@neverprocess\pstool@alwaysprocess
\LetLtxMacro\pstool@maybeprocess\pstool@alwaysprocess

\renewcommand\pstool@@psfragfig[4]{%
    \begin{lateximage}[-pstool-~\PackageDiagramAltText]%
    \includegraphics{#2.pdf}%
    \end{lateximage}%
}
```

File 398 **lwarp-pstricks.sty**

§ 507 Package pstricks

(*Emulates or patches code by* Timothy Van Zandt.)

pstricks (*Pkg*) pstricks is patched for use by lwarp.

⚠ use pspicture All pstricks content should be contained inside a pspicture environment.

for HTML output:

```
\LWR@ProvidesPackagePass{pstricks}[2018/01/06]

\BeforeBeginEnvironment{pspicture}{%
```

```
    \begin{lateximage}[pspicture]%
}
\AfterEndEnvironment{pspicture}{\end{lateximage}}

\BeforeBeginEnvironment{pspicture*}{%
    \begin{lateximage}[pspicture]%
}
\AfterEndEnvironment{pspicture*}{\end{lateximage}}
```

File 399 **lwarp-pxatbegshi.sty**

§ 508 Package pxatbegshi

pxatbegshi (*Pkg*) pxatbegshi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxatbegshi}[2017/11/04]

\LWR@origRequirePackage{lwarp-atbegshi}
```

File 400 **lwarp-pxeveryshi.sty**

§ 509 Package pxeveryshi

pxeveryshi (*Pkg*) pxeveryshi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxeveryshi}[2012/05/19]

\LWR@origRequirePackage{lwarp-everyshi}
```

File 401 **lwarp-pxfonts.sty**

§ 510 Package pxfonts

(*Emulates or patches code by* Young Ryu.)

pxfonts (*Pkg*) pxfonts is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{pxfonts}[2008/01/22]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{pxfonts}

\LWR@mathjax@addgreek@l@up{}{up}
\end{warpMathJax}
```

File 402 **lwarp-pxftnright.sty**

§ 511 Package pxftnright

pxftnright (*Pkg*) pxftnright is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxftnright}[2017/02/28]
\LWR@origRequirePackage{lwarp-ftnright}
```

File 403 **lwarp-pxjahyper.sty**

§ 512 Package pxjahyper

pxjahyper (*Pkg*) pxjahyper is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxjahyper}[2018/07/15]
```

File 404 **lwarp-quotchap.sty**

§ 513 Package quotchap

(*Emulates or patches code by* Karsten Tinnefeld, Jan Klever.)

quotchap (*Pkg*) quotchap is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{quotchap}[2019/07/09]

\newcommand{\@quotchap}{}
\newlength{\LWR@quotchapwidth}

\let\@printcites\relax

\newcommand*{\@iprintcites}{%
```

Place the quotes inside a of class quotchap, of the maximum selected width:

```
\begin{BlockClass}[max-width: \LWR@printlength{\LWR@quotchapwidth}]{quotchap}
%\begin{minipage}{\LWR@quotchapwidth}
\@quotchap
%\end{minipage}
\end{BlockClass}
```

Deactivate the quote printing:

```
\global\let\@printcites\relax
}

\NewEnviron{savequote}[1][\linewidth]{%
```

Remember the width, adjusted for HTML, and make the length assignment global, per:
https://tex.stackexchange.com/questions/300823/why-is-setlength-ineffective-inside-a-tabular-environment

```
\setlength{\LWR@quotchapwidth}{#1*2}%
\global\LWR@quotchapwidth=\LWR@quotchapwidth%
```

Remember the body, and activate the quote printing:

```
\global\let\@quotchap\BODY
\global\let\@printcites\@iprintcites%
}
```

The quotation author is placed inside a of class qauthor:

```
\newcommand{\qauthor}[1]{%
    \LWR@stoppars%
    \begin{BlockClass}{qauthor}%
    {#1}%
    \end{BlockClass}%
    \LWR@startpars%
}
```

Fonts are ignored. Use css.

```
\newcommand{\qsetcnfont}[1]{}
\providecommand*{\quotefont}{}
\providecommand*{\qauthorfont}{}
```

File 405 **lwarp-quoting.sty**

§ 514 Package quoting

(*Emulates or patches code by* Thomas Titz.)

quoting (*Pkg*) quoting is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{quoting}[2014/01/28]

\xpatchcmd{\quoting}{\quo@begintext}
    {\begin{LWR@blocktextcurrentfont}\quo@begintext}
    {}
    {\LWR@patcherror{quoting}{quoting}}

\xpatchcmd{\endquoting}{\quo@endtext}
    {\quo@endtext\end{LWR@blocktextcurrentfont}\LWR@stoppars}
    {}
    {\LWR@patcherror{quoting}{endquoting}}
```

File 406 **lwarp-ragged2e.sty**

§ 515 Package ragged2e

(*Emulates or patches code by* Martin Schröder.)

ragged2e (*Pkg*) ragged2e is emulated.

Discard all options for lwarp-ragged2e:

for HTML output:

```
\LWR@ProvidesPackageDrop{ragged2e}[2009/05/21]

\LetLtxMacro\Centering\centering
\LetLtxMacro\RaggedLeft\raggedleft
\LetLtxMacro\RaggedRight\raggedright
\newcommand*{\justifying}{}
\newlength{\CenteringLeftskip}
\newlength{\RaggedLeftLeftskip}
\newlength{\RaggedRightLeftskip}
\newlength{\CenteringRightskip}
\newlength{\RaggedLeftRightskip}
\newlength{\RaggedRightRightskip}
\newlength{\CenteringParfillskip}
\newlength{\RaggedLeftParfillskip}
\newlength{\RaggedRightParfillskip}
\newlength{\JustifyingParfillskip}
\newlength{\CenteringParindent}
\newlength{\RaggedLeftParindent}
\newlength{\RaggedRightParindent}
\newlength{\JustifyingParindent}
\newenvironment*{Center}{\center}{\endcenter}
\newenvironment*{FlushLeft}{\flushleft}{\endflushleft}
\newenvironment*{FlushRight}{\flushright}{\endflushright}
\newenvironment*{justify}{\justifying}{\endjustifying}
```

File 407 **lwarp-realscripts.sty**

§ 516 Package realscripts

(*Emulates or patches code by* Will Robertson.)

realscripts (*Pkg*) realscripts is emulated. See `lwarp.css` for the of class supsubscript.

for HTML output:

```
\LWR@ProvidesPackagePass{realscripts}[2016/02/13]

\ExplSyntaxOn

\DeclareDocumentCommand \LWR@HTML@realsubscript {m} {
    \LWR@HTML@textsubscript{#1}
}

\LWR@formatted{realsubscript}


\DeclareDocumentCommand \LWR@HTML@realsuperscript {m} {
    \LWR@HTML@textsuperscript{#1}
}

\LWR@formatted{realsuperscript}


\ExplSyntaxOff

```

```
\newcommand*{\LWR@realscriptsalign}{}

\newcommand*{\LWR@setrealscriptsalign}[1]{%
    \renewcommand*{\LWR@realscriptsalign}{}%
    \ifthenelse{\equal{#1}{c}}{%
        \renewcommand{\LWR@realscriptsalign}{%
            \LWR@print@mbox{text-align:center} ; %
        }%
    }{}%
    \ifthenelse{\equal{#1}{r}}{%
        \renewcommand{\LWR@realscriptsalign}{%
            \LWR@print@mbox{text-align:right} ; %
        }%
    }{}%
}

\DeclareDocumentCommand \LWR@HTML@textsubsuperscript {s O{l} mm} {%
    \LWR@setrealscriptsalign{#2}%
    \InlineClass[\LWR@realscriptsalign]{supsubscript}{%
        \textsuperscript{#4}\textsubscript{#3}%
    }%
}
\LWR@formatted{textsubsuperscript}

\FilenameNullify{%
    \RenewDocumentCommand{\textsuperscript}{s m}{}%
    \RenewDocumentCommand{\textsubscript}{s m}{}%
    \renewcommand{\fakesubscript}[1]{}%
    \renewcommand{\fakesuperscript}[1]{}%
    \renewcommand{\realsubscript}[1]{}%
    \renewcommand{\realsuperscript}[1]{}%
    \renewcommand{\textsubsuperscript}[2]{}%
    \renewcommand{\textsupersubscript}[2]{}%
}
```

File 408 **lwarp-refcheck.sty**

§ 517 Package **refcheck**

refcheck (*Pkg*) refcheck is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{refcheck}[2013/02/14]

\def\showrefnames{}
\def\norefnames{}
\def\showcitenames{}
\def\nocitenames{}
\def\setonmsgs{}
\def\setoffmsgs{}
\def\checkunlbld{}
\def\ignoreunlbld{}
\newcommand*{\refcheckxrdoc}[2][]{}
```

File 409 **lwarp-register.sty**

§ 518 Package register

(*Emulates or patches code by* Matthew Lovell.)

`register` (*Pkg*) register is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{register}[2019/01/01]

\xpatchcmd{\register}
    {\centering}
    {%
        \begin{center}%
        \begin{lateximage}[-register-~\PackageDiagramAltText]%
    }
    {}
    {\LWR@patcherror{register}{register}}

\xpatchcmd{\endregister}
    {\leftskip}
    {%
        \end{lateximage}\end{center}%
        \leftskip%
    }%
    {}
    {\LWR@patcherror{register}{endregister}}

\expandafter\xapptocmd\csname register*\endcsname
    {%
        \begin{center}%
        \begin{lateximage}[-register-~\PackageDiagramAltText]%
    }
    {}
    {\LWR@patcherror{register}{register*}}

\expandafter\xpatchcmd\csname endregister*\endcsname
    {\leftskip}
    {%
        \end{lateximage}\end{center}%
        \leftskip%
    }%
    {}
    {\LWR@patcherror{register}{endregister*}}

\setlength{\regWidth}{5in}
```

File 410 **lwarp-relsize.sty**

§ 519 Package relsize

(*Emulates or patches code by* Donald Arseneau, Bernie Cosell, Matt Swift.)

`relsize` (*Pkg*) relsize is patched for use by lwarp, and emulated for MathJax.

For HTML, only the inline macros are supported: `\textlarger`, `\textsmaller`, and `\textscale`. Each becomes an inline span of a modified `font-size`.

`\relsize`, `\larger`, `\smaller`, and `\relscale` are ignored.

While creating SVG math for HTML, the original definitions are temporarilty restored, and so should work as expected.

⚠ not small The HTML browser's setting for minumum font size may limit how small the output will be displayed.

for HTML output:

```
\LWR@ProvidesPackagePass{relsize}[2013/03/29]

\let\LWR@origrelsize\relsize
\LetLtxMacro\LWR@origlarger\larger
\LetLtxMacro\LWR@origsmaller\smaller
\let\LWR@relscale\relscale
\LetLtxMacro\LWR@origtextlarger\textlarger
\LetLtxMacro\LWR@origtextsmaller\textsmaller
\let\LWR@textscale\textscale

\appto\LWR@restoreorigformatting{%
\let\relsize\LWR@origrelsize%
\LetLtxMacro\larger\LWR@origlarger%
\LetLtxMacro\smaller\LWR@origsmaller%
\let\relscale\LWR@relscale%
\LetLtxMacro\textlarger\LWR@origtextlarger%
\LetLtxMacro\textsmaller\LWR@origtextsmaller%
\let\textscale\LWR@textscale%
}

\newcounter{LWR@relsizetemp}

\renewcommand*{\relsize}[1]{}
\renewcommand*{\larger}[1][]{}
\renewcommand*{\smaller}[1][]{}
\renewcommand*{\relscale}[1]{}

\renewcommand*{\textlarger}[2][1]{%
\setcounter{LWR@relsizetemp}{100+(#1*20)}%
\InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textlarger}{#2}%
}

\renewcommand*{\textsmaller}[2][1]{%
\setcounter{LWR@relsizetemp}{100-(#1*20)}%
\InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textsmaller}{#2}%
}

\renewcommand*{\textscale}[2]{%
\setcounter{LWR@relsizetemp}{100*\real{#1}}%
\InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textscale}{#2}%
}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mathlarger}[1]{#1}}
\CustomizeMathJax{\newcommand{\mathsmaller}[1]{#1}}
```

```
\end{warpMathJax}
```

File 411 **lwarp-repeatindex.sty**

§ 520 Package repeatindex

repeatindex (*Pkg*) repeatindex is emulated for lwarp.

⚠ style file lwarp must be used with a special style file:

```
\usepackage[makeindex,makeindexStyle={lwarp_repeatindex}]{lwarp}
```

where `lwarp_repeatindex.ist` may be copied from the following modified version of `lwarp.ist`:

```
preamble
"\\begin{theindex}
  \\providecommand*\\lettergroupDefault[1]{}
  \\providecommand*\\lettergroup[1]{%
      \\par\\textbf{#1}\\par
      \\nopagebreak
  }
"
headings_flag 1
heading_prefix "
  \\lettergroup{"
heading_suffix "}"
delim_0 "], \\hyperindexref{"
delim_1 ", \\hyperindexref{"
delim_2 ", \\hyperindexref{"
delim_n "}, \\hyperindexref{"
delim_r "} -- \\hyperindexref{"
delim_t "}"

item_0 "\n  \\item ["
```

(The modifications are the `delim_0` and `item_0` entries.)

for HTML output:

```
\LWR@ProvidesPackageDrop{repeatindex}[2001/10/13]
```

In the lwarp core, `\LWR@indexitem` is modified to accept the optional `\item` argument.

```
\RequirePackage{makeidx}
\def\entryprefix{\itshape}
\def\entrypostfix{\dots}
```

File 412 **lwarp-repltext.sty**

§ 521 Package repltext

repltext (*Pkg*) repltext is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{repltext}[2020/09/25]

\newcommand{\repltext}[2]{#2}
\newcommand*{\prevrepl}{}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\repltext}[2]{#2}}
\end{warpMathJax}
```

File 413 **lwarp-resizegather.sty**

§ 522 Package resizegather

`resizegather` (*Pkg*) resizegather is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{resizegather}[2016/05/16]

\newcommand*{\resizegathersetup}[1]{}
```

File 414 **lwarp-returntogrid.sty**

§ 523 Package returntogrid

`returntogrid` (*Pkg*) returntogrid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{returntogrid}[2018/08/21]

\NewDocumentCommand\returntogrid{ O {} }{}
\NewDocumentCommand\returntogridsetup { m } {}
\NewDocumentCommand\showdebugpagegrid {} {}
```

File 415 **lwarp-rlepsf.sty**

§ 524 Package rlepsf

(*Emulates or patches code by* Michael Greene, Colin Rourke.)

`rlepsf` (*Pkg*) rlepsf is patched for use by lwarp.

⚠ Rename the style file! The file `rlepsf.tex` must be copied to `rlepsf.sty` for lwarp to detect and patch it.

for HTML output:

```
\LWR@ProvidesPackagePass{rlepsf}% No date given.

\xpretocmd{\relabelbox}
    {\begin{lateximage}}
    {}
    {\LWR@patcherror{rlepsf}{relabelbox}}
```

```
6
7 \xapptocmd{\endrelabelbox}
8     {\end{lateximage}}
9     {}
10    {\LWR@patcherror{rlepsf}{endrelabelbox}}
```

File 416 **lwarp-rmathbr.sty**

§ 525 Package rmathbr

(*Emulates or patches code by* Denis Ryabov.)

rmathbr (*Pkg*) rmathbr is used as-is for svg math, and emulated for MathJax.

for HTML output:

```
1 \LWR@ProvidesPackagePass{rmathbr}[2020/12/11]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\def\*{~}}
4 \CustomizeMathJax{\newcommand{\cdott}{\cdot}}
5 \CustomizeMathJax{\newcommand{\nobr}{}}
6 \end{warpMathJax}
```

File 417 **lwarp-rmpage.sty**

§ 526 Package rmpage

rmpage (*Pkg*) rmpage is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{rmpage}[1997/09/29]
```

File 418 **lwarp-romanbar.sty**

§ 527 Package romanbar

(*Emulates or patches code by* H.-Martin Münch.)

romanbar (*Pkg*) romanbar is patched for use by lwarp.

An inline class with an overline and underline is used.

for HTML output:

```
1 \LWR@ProvidesPackagePass{romanbar}[2012/01/01]

2 \DeclareRobustCommand{\Roman@bar}[1]{% #1 is in Roman, i.e. MMXII
3 \InlineClass[%
4     text-decoration: overline underline ;
5 ]{romanbar}{#1}%
6 }
```

File 419 **lwarp-romanbarpagenumber.sty**

§ 528 Package
# romanbarpagenumber

romanbarpagenumber (*Pkg*) romanbarpagenumber is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{romanbarpagenumber}[2015/02/06]
```

File 420 **lwarp-rotating.sty**

§ 529 Package
# rotating

(*Emulates or patches code by* Robin Fairbairns, Sebastian Rahtz, Leonor Barroca.)

rotating (*Pkg*) rotating is emulated.

All rotations are ignored in HTML output.

for HTML output:

```
\LWR@ProvidesPackagePass{rotating}[2016/08/11]
\RequirePackage{graphicx}

\LetLtxMacro\LWR@HTML@sidewaystable\table
\let\endLWR@HTML@sidewaystable\endtable
\LWR@formattedenv{sidewaystable}

\LetLtxMacro\LWR@HTML@sidewaysfigure\figure
\let\endLWR@HTML@sidewaysfigure\endfigure
\LWR@formattedenv{sidewaysfigure}

\newenvironment*{LWR@HTML@sideways}{}{}
\LWR@formattedenv{sideways}

\newenvironment*{LWR@HTML@turn}[1]{}{}
\LWR@formattedenv{turn}

\newenvironment*{LWR@HTML@rotate}[1]{}{}
\LWR@formattedenv{rotate}

\NewDocumentCommand{\LWR@HTML@turnbox}{m +m}{#2}
\LWR@formatted{turnbox}

\let\LWR@HTML@rotcaption\caption
\LWR@formatted{rotcaption}

\let\LWR@HTML@@makerotcaption\@makecaption
\LWR@formatted{@makerotcaption}
```

File 421 **lwarp-rotfloat.sty**

§ 530 Package rotfloat

(*Emulates or patches code by* Axel Sommerfeldt.)

`rotfloat` (*Pkg*) rotfloat is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{rotfloat}[2004/01/04]

\RequirePackage{float}
\RequirePackage{rotating}
```

`\newfloat` {⟨*1: type*⟩} {⟨*2: placement*⟩} {⟨*3: ext*⟩} [⟨*4: within*⟩]

Emulates the `\newfloat` command from the float package. Sideways floats are `\let` to the same as regular floats.

"placement" is ignored.

```
\RenewDocumentCommand{\newfloat}{m m m o}{%
\IfValueTF{#4}%
{%
    \DeclareFloatingEnvironment[fileext=#3,within=#4]{#1}%
}%
{%
    \DeclareFloatingEnvironment[fileext=#3]{#1}%
}%
\csletcs{sideways#1}{#1}%
\csletcs{endsideways#1}{end#1}%
```

Remember the float style:

```
\csedef{LWR@floatstyle@#1}{\LWR@floatstyle}%
\csedef{LWR@floatstyle@sideways#1}{\LWR@floatstyle}%
```

newfloat package automatically creates the `\listof` command for new floats, but float does not, so remove `\listof` here in case it is manually created later:

```
\cslet{listof#1s}\relax%
\cslet{listof#1es}\relax%
\cslet{listofsideways#1s}\relax%
\cslet{listofsideways#1es}\relax%
}
```

File 422 **lwarp-rviewport.sty**

§ 531 Package rviewport

`rviewport` (*Pkg*) rviewport is honored inside a `lateximage`, and otherwise ignored for HTML output.

If rviewport is important for an image, enclose the image inside a `lateximage` environment.

for HTML output:

```
\LWR@ProvidesPackagePass{rviewport}[2011/08/27]
\define@key{igraph}{rviewport}{}
```

File 423 **lwarp-savetrees.sty**

§ 532 Package savetrees

savetrees (*Pkg*) savetrees is ignored.

for HTML output: Discard all options for lwarp-savetrees:

```
\LWR@ProvidesPackageDrop{savetrees}[2016/04/13]
```

File 424 **lwarp-scalefnt.sty**

§ 533 Package scalefnt

(*Emulates or patches code by* D. Carlisle.)

scalefnt (*Pkg*) scalefnt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{scalefnt}
\DeclareRobustCommand\scalefont[1]{}
```

File 425 **lwarp-scalerel.sty**

§ 534 Package scalerel

(*Emulates or patches code by* Steven B. Segletes.)

scalerel (*Pkg*) scalerel is used as-is for SVG math, and is emulated and ignored for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{scalerel}[2016/12/29]
```

For MathJax:

```
\begin{warpMathJax}
\LWR@infoprocessingmathjax{scalerel}

\CustomizeMathJax{\newcommand{\scalerel}{\ifstar{\scalerelplain}{\scalerelplus}}}
\CustomizeMathJax{\newcommand{\scalerelplain}[3][]{#2}}
\CustomizeMathJax{\newcommand{\scalerelplus}[3][]{#2#3}}
\CustomizeMathJax{\newcommand{\stretchrel}{\ifstar{\stretchrelplain}{\stretchrelplus}}}
\CustomizeMathJax{\newcommand{\stretchrelplain}[3][]{#2}}
\CustomizeMathJax{\newcommand{\stretchrelplus}[3][]{#2#3}}
\CustomizeMathJax{\newcommand{\scaleto}[3][]{#2}}
```

```
\CustomizeMathJax{\newcommand{\stretchto}[3][]{#2}}
\CustomizeMathJax{\newcommand{\scaleleftright}[4][]{#2#3#4}}
\CustomizeMathJax{\newcommand{\stretchleftright}[4][]{#2#3#4}}
\CustomizeMathJax{\newcommand{\hstretch}[2]{#2}}
\CustomizeMathJax{\newcommand{\vstretch}[2]{#2}}
\CustomizeMathJax{\newcommand{\scaleobj}[2]{#2}}
\CustomizeMathJax{\newcommand{\ThisStyle}[1]{#1}}
\CustomizeMathJax{\newcommand{\SavedStyle}{}}
\CustomizeMathJax{\def\scriptstyleScaleFactor{.7}}
\CustomizeMathJax{\def\scriptscriptstyleScaleFactor{.5}}
\CustomizeMathJax{\newcommand{\discernmathstyle}{}}
\CustomizeMathJax{\newcommand{\ignoremathstyle}[1][T]{}}
\CustomizeMathJax{\newcommand{\Isnextbyte}[3][v]{}}
\end{warpMathJax}
```

File 426 **lwarp-schemata.sty**

§ 535 Package schemata

(*Emulates or patches code by* Charles P. Schaum.)

schemata (*Pkg*) schemata is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{schemata}[2020/11/23]

\LetLtxMacro\LWR@schemata@origschema\schema
\LetLtxMacro\LWR@schemata@origSchema\Schema

\renewcommand{\schema}[3][open]{%
    \begin{lateximage}[-schemata-~\PackageDiagramAltText]%
    \LWR@print@normalsize%
    \LWR@schemata@origschema[#1]{#2}{#3}%
    \end{lateximage}%
}

\renewcommand{\Schema}[5][open]{%
    \begin{lateximage}[-schemata-~\PackageDiagramAltText]%
    \LWR@print@normalsize%
    \LWR@schemata@origSchema[#1]{#2}{#3}{#4}{#5}%
    \end{lateximage}%
}
```

File 427 **lwarp-scrextend.sty**

§ 536 Package scrextend

scrextend (*Pkg*) scrextend is emulated.

This package may be loaded standalone, but is also loaded automatically if koma-script classes are in use. \DeclareDocumentCommand is used to overwrite the koma-script definitions.

for HTML output:

```
\LWR@ProvidesPackageDrop{scrextend}[2020/01/24]
```

```
\DeclareDocumentCommand{\setkomafont}{m m}{}
\DeclareDocumentCommand{\addkomafont}{m m}{}
\DeclareDocumentCommand{\usekomafont}{m}{}

\DeclareDocumentCommand{\usefontofkomafont}{m}{}
\DeclareDocumentCommand{\useencodingofkomafont}{m}{}
\DeclareDocumentCommand{\usesizeofkomafont}{m}{}
\DeclareDocumentCommand{\usefamilyofkomafont}{m}{}
\DeclareDocumentCommand{\useseriesofkomafont}{m}{}
\DeclareDocumentCommand{\useshapeofkomafont}{m}{}

\providecommand*{\coverpagetopmargin}{}
\providecommand*{\coverpagebottommargin}{}
\providecommand*{\coverpageleftmargin}{}
\providecommand*{\coverpagerightmargin}{}

```

Title page:

```
\AtBeginDocument{
    \let\LWR@koma@orig@maketitle\maketitle
    \DeclareDocumentCommand{\maketitle}{o}{\LWR@koma@orig@maketitle}
}

\providecommand*{\@maketitle}{}
\renewrobustcmd{\@maketitle}{%
    \ifdefvoid{\@titlehead}{}{%
        \begin{BlockClass}{titlehead}%
        \@titlehead%
        \end{BlockClass}%
    }%
    \ifdefvoid{\@subject}{}{%
        \begin{BlockClass}{subject}%
        \@subject%
        \end{BlockClass}%
    }%
    \LWR@stoppars%
    \LWR@htmltag{\LWR@tagtitle}%
    \@title%
    \LWR@htmltag{\LWR@tagtitleend}%
    \ifdefvoid{\@subtitle}{}{%
        \begin{BlockClass}{subtitle}%
        \@subtitle%
        \end{BlockClass}%
    }%
    \LWR@startpars%
    \begin{BlockClass}{author}%

    \renewcommand*{\cr}{}%
    \renewcommand*{\crcr}{}%
    \renewcommand*{\noalign}{}%

        \renewcommand{\and}{%
            \end{BlockClass}%
            \begin{BlockClass}{oneauthor}%
        }%
        \begin{BlockClass}{oneauthor}%
            \@author%
        \end{BlockClass}%
```

```
    \end{BlockClass}%
    \begin{BlockClass}{titledate}%
    \@date%
    \end{BlockClass}%
    \ifdefvoid{\@published}{}{%
        \begin{BlockClass}{published}%
        \@published%
        \end{BlockClass}%
    }%
}

\AddSubtitlePublished

\DeclareDocumentCommand{\extratitle}{m}{}
\DeclareDocumentCommand{\frontispiece}{m}{}

\def\@titlehead{}%
\DeclareDocumentCommand{\titlehead}{m}{\gdef\@titlehead{#1}}%

\def\@subject{}%
\DeclareDocumentCommand{\subject}{m}{\gdef\@subject{#1}}%

% \subtitle and \published are defined by \AddSubtitlePublished

\DeclareDocumentCommand{\publishers}{m}{\published{#1}}

\DeclareDocumentCommand{\uppertitleback}{m}{}
\DeclareDocumentCommand{\lowertitleback}{m}{}
\DeclareDocumentCommand{\dedication}{m}{}

\DeclareDocumentCommand{\ifthispageodd}{m m}{#1}

\DeclareDocumentCommand{\cleardoublepageusingstyle}{m}{}
\DeclareDocumentCommand{\cleardoubleemptypage}{}{}
\DeclareDocumentCommand{\cleardoubleplainpage}{}{}
\DeclareDocumentCommand{\cleardoublestandardpage}{}{}
\DeclareDocumentCommand{\cleardoubleoddpage}{}{}
\DeclareDocumentCommand{\cleardoubleoddpageusingstyle}{m}{}
\DeclareDocumentCommand{\cleardoubleoddemptypage}{}{}
\DeclareDocumentCommand{\cleardoubleoddplainpage}{}{}
\DeclareDocumentCommand{\cleardoubleoddstandardpage}{}{}
\DeclareDocumentCommand{\cleardoubleevenpage}{}{}
\DeclareDocumentCommand{\cleardoubleevenpageusingstyle}{m}{}
\DeclareDocumentCommand{\cleardoubleevenemptypage}{}{}
\DeclareDocumentCommand{\cleardoubleevenplainpage}{}{}
\DeclareDocumentCommand{\cleardoubleevenstandardpage}{}{}

\DeclareDocumentCommand{\multiplefootnoteseparator}{}{%
  \begingroup\let\thefootnotemark\multfootsep\@makefnmark\endgroup
}

\DeclareDocumentCommand{\multfootsep}{}{,}

\DeclareDocumentCommand{\footref}{m}{%
  \begingroup
    \unrestored@protected@xdef\@thefnmark{\ref{#1}}%
  \endgroup
  \@footnotemark
}

```

```
\DeclareDocumentCommand{\deffootnote}{o m m m}{}
\DeclareDocumentCommand{\deffootnotemark}{m}{}
\DeclareDocumentCommand{\setfootnoterule}{o m}{}
\DeclareDocumentCommand{\raggedfootnote}{}{}


\DeclareDocumentCommand{\dictum}{o m}{
\begin{LWR@BlockClassWP}{\LWR@print@mbox{text-align:right}}{}{dictum}
    #2
    \IfValueT{#1}
    {
        \LWR@stoppars%
        \ifbool{FormatWP}
     {\begin{BlockClass}[\LWR@print@mbox{border-top:} 1px solid gray]{dictumauthor}}
        {\begin{BlockClass}{dictumauthor}}
        \dictumauthorformat{#1}
        \end{BlockClass}
    }
\end{LWR@BlockClassWP}
}

\DeclareDocumentCommand{\dictumwidth}{}{}
\DeclareDocumentCommand{\dictumauthorformat}{m}{(#1)}
\DeclareDocumentCommand{\dictumrule}{}{}
\DeclareDocumentCommand{\raggeddictum}{}{}
\DeclareDocumentCommand{\raggeddictumtext}{}{}
\DeclareDocumentCommand{\raggeddictumauthor}{}{}

\DeclareDocumentEnvironment{labeling}{o m}
{%
\def\sc@septext{#1}%
\list{}{}%
\let\makelabel\labelinglabel%
}
{
\endlist
}

\DeclareDocumentCommand{\labelinglabel}{m}{%
#1 \qquad \sc@septext%
}

\let\addmargin\relax
\let\endaddmargin\relax
\cslet{addmargin*}{\relax}
\cslet{endaddmargin*}{\relax}


\NewDocumentEnvironment{addmargin}{s O{} m}
{
\LWR@stoppars%
\setlength{\LWR@templengthtwo}{#3}
\ifblank{#2}
{
    \begin{BlockClass}[
        \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthtwo}} ;
        \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}
    ]{addmargin}
}
{
    \setlength{\LWR@templengthone}{#2}
```

```
    \begin{BlockClass}[
        \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ;
        \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}
    ]{addmargin}
}
}
{\end{BlockClass}\LWR@startpars}
```

Ref to create a starred environment:
https://tex.stackexchange.com/questions/45401/use-the-s-star-argument-with-newdocumentenvironment

```

\ExplSyntaxOn
\cs_new:cpn {addmargin*} {\addmargin*}
\cs_new_eq:cN {endaddmargin*} \endaddmargin
\ExplSyntaxOff

\DeclareDocumentCommand{\marginline}{m}{\marginpar{#1}}
```

File 428 **lwarp-scrhack.sty**

§ 537 Package scrhack

scrhack (*Pkg*) scrhack is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{scrhack}[2018/03/30]
```

File 429 **lwarp-scrlayer.sty**

§ 538 Package scrlayer

(*Emulates or patches code by* Markus Kohm.)

scrlayer (*Pkg*) scrlayer is emulated.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrlayer}[2018/03/30]

\newcommand*{\DeclareSectionNumberDepth}[2]{}
\newcommand*{\DeclareLayer}[2][]{}
\newcommand*{\DeclareNewLayer}[2][]{}
\newcommand*{\ProvideLayer}[2][]{}
\newcommand*{\RedeclareLayer}[2][]{}
\newcommand*{\ModifyLayer}[2][]{}
\newcommand*{\layerhalign}{}
\newcommand*{\layervalign}{}
\newcommand*{\layerxoffset}{}
\newcommand*{\layeryoffset}{}
\newcommand*{\layerwidth}{}
\newcommand*{\layerheight}{}
\providecommand*{\LenToUnit}[1]{\strip@pt\dimexpr#1*\p@/\unitlength}
\newcommand*{\putUL}[1]{}
```

```
\newcommand*{\putUR}[1]{}
\newcommand*{\putLL}[1]{}
\newcommand*{\putLR}[1]{}
\newcommand*{\putC}[1]{}
\newcommand*{\GetLayerContents}[1]{}
\newcommand{\IfLayerExists}[3]{#3}
\newcommand*{\DestroyLayer}[1]{}
\newcommand*{\layercontentsmeasure}{}
\newcommand*{\currentpagestyle}{}
\newcommand*{\BeforeSelectAnyPageStyle}[1]{}
\newcommand*{\AfterSelectAnyPageStyle}[1]{}
\newcommand*{\DeclarePageStyleAlias}[2]{}
\newcommand*{\DeclareNewPageStyleAlias}[2]{}
\newcommand*{\ProvidePageStyleAlias}[2]{}
\newcommand*{\RedeclarePageStyleAlias}[2]{}
\newcommand*{\DestroyPageStyleAlias}[1]{}
\newcommand*{\GetRealPageStyle}[1]{}
\newcommand*{\DeclarePageStyleByLayers}[3][]{}
\newcommand*{\DeclareNewPageStyleByLayers}[3][]{}
\newcommand*{\ProvidePageStyleByLayers}[3][]{}
\newcommand*{\RedeclarePageStyleByLayers}[3][]{}
\NewDocumentCommand{\ForEachLayerOfPageStyle}{s m m}{}
\newcommand*{\AddLayersToPageStyle}[2]{}
\newcommand*{\AddLayersAtBeginOfPageStyle}[2]{}
\newcommand*{\AddLayersAtEndOfPageStyle}[2]{}
\newcommand*{\RemoveLayersFromPageStyle}[2]{}
\newcommand*{\AddLayersToPageStyleBeforeLayer}[3]{}
\newcommand*{\AddLayersToPageStyleAfterLayer}[3]{}
\newcommand*{\UnifyLayersAtPageStyle}[1]{}
\newcommand*{\ModifyLayerPageStyleOptions}[2]{}
\newcommand*{\AddToLayerPageStyleOptions}[2]{}
\newcommand{\IfLayerPageStyleExists}[3]{#3}
\newcommand{\IfRealLayerPageStyleExists}[3]{#3}
\newcommand{\IfLayerAtPageStyle}[4]{#4}
\newcommand{\IfSomeLayerAtPageStyle}[4]{#4}
\newcommand{\IfLayersAtPageStyle}[4]{#4}
\newcommand*{\DestroyRealLayerPageStyle}[1]{}
\@ifundefined{footheight}{\newlength\footheight}{}
\DeclareDocumentCommand{\automark}{s o m}{}
\DeclareDocumentCommand{\manualmark}{}{}
\DeclareDocumentCommand{\MakeMarkcase}{m}{#1}


\newcommand{\partmarkformat}{}
\if@chapter
\newcommand{\chaptermarkformat}{}
\fi
\newcommand{\sectionmarkformat}{}
\DeclareDocumentCommand{\GenericMarkFormat}{m}{}

\newcommand*{\@mkleft}[1]{}
\newcommand*{\@mkright}[1]{}
\newcommand*{\@mkdouble}[1]{}
\newcommand*{\@mkboth}[2]{}
\newcommand*{\scrlayerInitInterface}[1][]{}
\newcommand{\scrlayerAddToInterface}[3][]{}
\newcommand{\scrlayerAddCsToInterface}[3][]{}
\newcommand{\scrlayerOnAutoRemoveInterface}[2][]{}
```

File 430 **lwarp-scrlayer-notecolumn.sty**

§ 539 Package scrlayer-notecolumn

(*Emulates or patches code by* Markus Kohm.)

scrlayer-notecolumn (*Pkg*) scrlayer-notecolumn is emulated.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrlayer-notecolumn}[2018/02/02]
\newcommand*{\DeclareNoteColumn}[2][]{}
\newcommand*{\DeclareNewNoteColumn}[2][]{}
\newcommand*{\ProvideNoteColumn}[2][]{}
\newcommand*{\RedeclareNoteColumn}[2][]{}
\NewDocumentCommand{\makenote}{s o m}{\marginpar{#3}}
\newcommand*{\syncwithnotecolumn}[1][]{}
\newcommand*{\syncwithnotecolumns}[1][]{}
\newcommand*{\clearnotecolumn}[1][]{}
\newcommand*{\clearnotecolumns}[1][]{}
```

File 431 **lwarp-scrlayer-scrpage.sty**

§ 540 Package scrlayer-scrpage

(*Emulates or patches code by* Markus Kohm.)

scrlayer-scrpage (*Pkg*) scrlayer-scrpage is ignored.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrlayer-scrpage}[2018/03/30]
\@ifundefined{footheight}{\newlength\footheight}{}
\NewDocumentCommand{\lehead}{s o m}{}
\NewDocumentCommand{\cehead}{s o m}{}
\NewDocumentCommand{\rehead}{s o m}{}
\NewDocumentCommand{\lohead}{s o m}{}
\NewDocumentCommand{\cohead}{s o m}{}
\NewDocumentCommand{\rohead}{s o m}{}
\NewDocumentCommand{\lefoot}{s o m}{}
\NewDocumentCommand{\cefoot}{s o m}{}
\NewDocumentCommand{\refoot}{s o m}{}
\NewDocumentCommand{\lofoot}{s o m}{}
\NewDocumentCommand{\cofoot}{s o m}{}
\NewDocumentCommand{\rofoot}{s o m}{}
\NewDocumentCommand{\ohead}{s o m}{}
\NewDocumentCommand{\chead}{s o m}{}
\NewDocumentCommand{\ihead}{s o m}{}
\NewDocumentCommand{\ofoot}{s o m}{}
\NewDocumentCommand{\cfoot}{s o m}{}
\NewDocumentCommand{\ifoot}{s o m}{}
```

```
\NewDocumentCommand{\automark}{som}{}
\newcommand*{\manualmark}{}

\DeclareDocumentCommand{\MakeMarkcase}{m}{#1}

\let\headmark\leftmark
\providecommand{\pnumfont}{\normalfont}%
\DeclareRobustCommand\pagemark{{\pnumfont{\thepage}}}%

\newcommand*{\defpairofpagestyles}[3][]{}
\newcommand*{\newpairofpagestyles}[3][]{}
\newcommand*{\renewpairofpagestyles}[3][]{}
\newcommand*{\providepairofpagestyles}[3][]{}

\newcommand*{\clearmainofpairofpagestyles}{}
\newcommand*{\clearplainofpairofpagestyles}{}
\newcommand*{\clearpairofpagestyles}{}
\newcommand*{\clearscrheadings}{}
\newcommand*{\clearscrheadfoot}{}
\newcommand*{\clearscrplain}{}

\NewDocumentCommand{\deftriplepagestyle}{m o o m m m m m}{}
\NewDocumentCommand{\newtriplepagestyle}{m o o m m m m m}{}
\NewDocumentCommand{\renewtriplepagestyle}{m o o m m m m m}{}
\NewDocumentCommand{\providetriplepagestyle}{m o o m m m m m}{}
\newcommand*{\defpagestyle}[3]{}
\newcommand*{\newpagestyle}[3]{}
\newcommand*{\providepagestyle}[3]{}
\newcommand*{\renewpagestyle}[3]{}
```

File 432 **lwarp-scrpage2.sty**

§ 541 Package

scrpage2

(*Emulates or patches code by* Markus Kohm.)

scrpage2 (*Pkg*) scrpage2 is ignored.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrpage2}[2018/03/30]

\@ifundefined{footheight}{\newlength\footheight}{}
\NewDocumentCommand{\lehead}{o m}{}
\NewDocumentCommand{\cehead}{o m}{}
\NewDocumentCommand{\rehead}{o m}{}
\NewDocumentCommand{\lohead}{o m}{}
\NewDocumentCommand{\cohead}{o m}{}
\NewDocumentCommand{\rohead}{o m}{}
\NewDocumentCommand{\lefoot}{o m}{}
\NewDocumentCommand{\cefoot}{o m}{}
\NewDocumentCommand{\refoot}{o m}{}
\NewDocumentCommand{\lofoot}{o m}{}
\NewDocumentCommand{\cofoot}{o m}{}
\NewDocumentCommand{\rofoot}{o m}{}
```

```
\NewDocumentCommand{\ohead}{o m}{}
\NewDocumentCommand{\chead}{o m}{}
\NewDocumentCommand{\ihead}{o m}{}
\NewDocumentCommand{\ofoot}{o m}{}
\NewDocumentCommand{\cfoot}{o m}{}
\NewDocumentCommand{\ifoot}{o m}{}
\DeclareDocumentCommand{\automark}{o m}{}
\DeclareDocumentCommand{\manualmark}{}{}
\DeclareDocumentCommand{\MakeMarkcase}{m}{#1}
\NewDocumentCommand{\deftripstyle}{m o o m m m m m m}{}
\NewDocumentCommand{\defpagestyle}{s m m m}{}
\NewDocumentCommand{\newpagestyle}{s m m m}{}
\NewDocumentCommand{\renewpagestyle}{s m m m}{}
\NewDocumentCommand{\providepagestyle}{s m m m}{}
\newcommand{\partmarkformat}{}
\if@chapter
\newcommand{\chaptermarkformat}{}
\fi
\newcommand{\sectionmarkformat}{}
\newcommand{\subsectionmarkformat}{}
\newcommand{\subsubsectionmarkformat}{}
\newcommand{\paragraphmarkformat}{}
\newcommand{\subparagraphmarkformat}{}

\newcommand*{\clearscrheadings}{}
\newcommand*{\clearscrheadfoot}{}
\newcommand*{\clearscrplain}{}
```

File 433 **lwarp-section.sty**

§ 542 Package section

section (*Pkg*) section is ignored.

(*Emulates or patches code by* Oliver Pretzel.)

for HTML output:

```
\LWR@ProvidesPackageDrop{section}

\ifx\chapter\undefined
 \def\chsize{\Large}\def\hdsize{\huge}\else
 \def\chsize{\huge}\def\hdsize{\Huge}
\fi
\let\ttsize\LARGE
\let\ausize\large
\let\dasize\large
\let\secsize\Large
\let\subsize\large
\let\hdpos\raggedright
\newcounter{hddepth}
\let\fpind\relax
\def\ttfnt{}
\def\hdfnt{}
\def\fefnt{}
\def\thfnt{}
\def\pgfnt{}
\def\hmkfnt{}
\let\mkcse\uppercase
```

```
\def\hddot{}
\def\cpdot{:}
\def\nmdot{}
\ifx\secindent\undefined
 \newdimen\secindent
 \newskip\secpreskp
 \newskip\secpstskp
 \newdimen\subindent
 \newskip\subpreskp
 \newskip\subpstskp
 \newskip\parpstskp
 \newcount\c@hddepth
\fi
```

File 434 **lwarp-sectionbreak.sty**

§ 543 Package sectionbreak

(*Emulates or patches code by* Michal Hoftich.)

sectionbreak (*Pkg*) sectionbreak is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{sectionbreak}[2018-01-03]

\renewcommand\asterism{\HTMLunicode{2042}}

\renewcommand\pre@sectionbreak{}
\renewcommand\post@sectionbreak{}

\renewcommand\print@sectionbreak[1]{%
\begin{center}
#1
\end{center}
}

```

File 435 **lwarp-sectsty.sty**

§ 544 Package sectsty

(*Emulates or patches code by* Rowland McDonnell.)

sectsty (*Pkg*) sectsty is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{sectsty}[2002/02/25]

\newcommand*{\partfont}          [1] {}
\newcommand*{\partnumberfont}    [1] {}
\newcommand*{\parttitlefont}     [1] {}
\newcommand*{\chapterfont}       [1] {}
\newcommand*{\chapternumberfont} [1] {}
\newcommand*{\chaptertitlefont}  [1] {}
\newcommand*{\sectionfont}       [1] {}
\newcommand*{\subsectionfont}    [1] {}
```

```
\newcommand*{\subsubsectionfont} [1] {}
\newcommand*{\paragraphfont}     [1] {}
\newcommand*{\subparagraphfont}  [1] {}
\newcommand*{\minisecfont}  [1] {}
\newcommand*{\allsectionsfont}[1] {}
\newcommand{\nohang}{}
```

`\sectionrule` is only to be used in `*font` commands, thus it is ignored.

```
\newcommand*{\sectionrule}[5]{}

\def\ulemheading#1#2{}
```

File 436 **lwarp-selectp.sty**

§ 545 Package selectp

`selectp` (*Pkg*) selectp is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{selectp}% no date given

\newcommand*{\outputonly}[1]{}
```

File 437 **lwarp-semantic-markup.sty**

§ 546 Package semantic-markup

(*Emulates or patches code by* Andrew A. Cashner.)

`semantic-markup` (*Pkg*) semantic-markup is patched for use by lwarp.

⚠ If using the endnotes option, add `\theendnotes` where desired.

for HTML output:

```
\LWR@ProvidesPackagePass{semantic-markup}[2018/05/21]
```

The endnotes must be printed by the user before the end of the document, since the end is after the HTML footer, etc.

```
\ifendnotes
\RenewDocumentCommand{\SetupEndnotes}{}{%
    \let\footnote=\endnote
%     \AtEndDocument{\DoBeforeEndnotes{\EndnoteFont\theendnotes}}%
}
\fi
```

HTML unicode characters from musicography are used.

```
\RequirePackage{musicography}

\let\fl\musFlat
\let\sh\musSharp
\let\na\musNatural
```

The \musfig is placed inside a hashed image, with a simple alt tag.

```
\RequirePackage{amsmath}

\RenewDocumentCommand{\musfig}{ m m }{%
    \LWR@subsingledollar*%
        {#1/#2}% alt tag
        {musfig}% addl' hashing
        {% contents
            \LWR@origensuredmath{%
                \genfrac{}{}{0pt}{1}{\text{#1}}{\text{#2}}%
            }%
        }%
}
```

The \meter is taken from musicography, and becomes a hashed image with a simple alt tag.

```
\RenewDocumentCommand{\meter}{ m m }{%
    \musMeter{#1}{#2}%
}
```

File 438 **lwarp-seqsplit.sty**

§ 547 Package

seqsplit

(*Emulates or patches code by* Boris Veytsman.)

seqsplit (*Pkg*) seqsplit is patched for use by lwarp.

For HTML output, the results are similar to print mode, and respond to window size.

⚠ SVG math results For SVG math, the output differs from print mode in that the contents are formatted in a minipage, which is then inline with the surrounding math.

For MathJax, the contents are used as-is.

for HTML output:

```
\LWR@ProvidesPackagePass{seqsplit}[2006/08/07]
```

Special handling because lwarp uses a box for SVG math, which does not normally allow line breaks, so a print-mode minipage must be used to allow line breaks. The minipage will not be wrapped inline with any surrounding math.

```
\begin{warpHTML}
\LetLtxMacro\LWR@orig@seqsplit\seqsplit

\renewcommand*{\seqsplit}[1]{%
    \ifmmode%
        \begin{LWR@print@minipage}{6in}%
        \LWR@orig@seqsplit{#1}%
        \end{LWR@print@minipage}%
    \else%
      \InlineClass[word-wrap:break-word]{seqsplit}{\LWR@orig@seqsplit{#1}}%
    \fi
}
```

Between characters, an empty HTML comment is placed to allow a line wrap in the HTML source, without adding spaces in the output.

```
\AtBeginDocument{
    \newcommand*{\LWR@HTML@seqinsert}{%
        \LWR@htmlcomment{ }%
    }
    \LWR@formatted{seqinsert}
}
\end{warpHTML}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\seqsplit}[1]{#1}}
\end{warpMathJax}
```

File 439 **lwarp-setspace.sty**

§ 548 Package setspace

(*Emulates or patches code by* Robin Fairbairns.)

setspace (*Pkg*) setspace is emulated.

Discard all options for lwarp-setspace:

for HTML output:

```
\LWR@ProvidesPackageDrop{setspace}[2011/12/19]

\newcommand*{\setstretch}[1]{}
\newcommand*{\SetSinglespace}[1]{}
\newcommand*{\singlespacing}{}
\newcommand*{\onehalfspacing}{}
\newcommand*{\doublespacing}{}

\newenvironment*{singlespace}
{
\LWR@forcenewpage
\BlockClass{singlespace}
}
{\endBlockClass}

\newenvironment*{singlespace*}
{
\LWR@forcenewpage
\BlockClass{singlespace}
}
{\endBlockClass}

\newenvironment*{spacing}[1]{

}{

}

\newenvironment*{onehalfspace}
{
\LWR@forcenewpage
\BlockClass{onehalfspace}
}
```

```
{\endBlockClass}

\newenvironment*{doublespace}
{
\LWR@forcenewpage
\BlockClass{doublespace}
}
{\endBlockClass}
```

File 440 **lwarp-shadethm.sty**

§ 549 Package shadethm

(*Emulates or patches code by* Jim Hefferon.)

shadethm (*Pkg*) shadethm is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{shadethm}[1999/11/23]

\newenvironment{LWR@HTML@shadebox}
{%
    \convertcolorspec{named}{shadethmcolor}{HTML}\LWR@tempcolor%
    \convertcolorspec{named}{shaderulecolor}{HTML}\LWR@tempcolortwo%
    \begin{BlockClass}[%
        background: \LWR@origpound\LWR@tempcolor ;
        border: 1px solid \LWR@origpound\LWR@tempcolortwo ;
    ]{shadebox}
}%
{\end{BlockClass}}
\LWR@formattedenv{shadebox}
```

File 441 **lwarp-shadow.sty**

§ 550 Package shadow

(*Emulates or patches code by* Mauro Orlandini.)

shadow (*Pkg*) shadow is emulated.

for HTML output: Discard all options for lwarp-shadow:

```
\LWR@ProvidesPackageDrop{shadow}[2003/02/19]

\newdimen\sboxsep
\newdimen\sboxrule
\newdimen\sdim

\newcommand{\shabox}[1]{%
\InlineClass{shabox}{#1}%
}
```

File 442 **lwarp-shapepar.sty**

§ 551 Package shapepar

(*Emulates or patches code by* Donald Arseneau.)

shapepar (*Pkg*) shapepar is patched for use by lwarp. Shapes appear in print mode, as well as inside a `lateximage`, but are ignored for HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{shapepar}[2013/03/26]

\newcommand*{\LWR@HTML@shapepar}[2][]{}
\LWR@formatted{shapepar}

\NewDocumentCommand{\LWR@HTML@cutout}{m d()}{}
\LWR@formatted{cutout}
```

File 443 **lwarp-showidx.sty**

§ 552 Package showidx

showidx (*Pkg*) showidx is ignored.

for HTML output: Discard all options for lwarp-showidx:

```
\LWR@ProvidesPackageDrop{showidx}[2014/09/29]
```

`\@wrindex` is redefined `\AtBeginDocument` by the lwarp core.

File 444 **lwarp-showkeys.sty**

§ 553 Package showkeys

(*Emulates or patches code by* David Carlisle, Morten Høgholm.)

showkeys (*Pkg*) showkeys is ignored.

for HTML output: Discard all options for lwarp-showkeys:

```
\LWR@ProvidesPackageDrop{showkeys}[2014/10/28]

\NewDocumentCommand{\showkeys}{s}{}
```

File 445 **lwarp-showlabels.sty**

§ 554 Package showlabels

showlabels (*Pkg*) showlabels is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{showlabels}[2021/10/27]

\providecommand{\showlabelfont}{}
\providecommand{\showlabelsetlabel}[1]{}
\newcommand*{\showlabels}[2][]{}
\newcommand*{\showlabelrefline}{}
\newcommand*{\showlabelsinline}{}
```

File 446 **lwarp-showtags.sty**

§ 555 Package showtags

showtags (*Pkg*) showtags is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{showtags}% no version is given

\newcommand{\thecitetag}[1]{}
```

File 447 **lwarp-shuffle.sty**

§ 556 Package shuffle

(*Emulates or patches code by* Julian Gilbey and Antoine Lejay.)

shuffle (*Pkg*) shuffle is emulated for SVG math, and also emulated for MathJax.

The font used for shuffle may not render correctly when converted to SVG math, so a `picture` environment drawing is used instead.

For MathJax, the Unicode character is used, and for `\cshuffle` a `\bar` is added.

for HTML output:

```
\LWR@ProvidesPackageDrop{shuffle}[2008/10/27]
\LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}

\newcommand*{\LWR@shuffle@start}{%
    \hspace*{.2em}
    \begin{picture}(.75,0.65)
    \setlength{\unitlength}{1em}
    \put(0,0){\line(1,0){.75}}
    \put(0,0){\line(0,1){.5}}
    \put(.375,0){\line(0,1){.5}}
    \put(.75,0){\line(0,1){.5}}
}

\newcommand*{\LWR@shuffle@finish}{%
    \end{picture}
    \hspace*{.75em}
    \hspace*{.2em}
}

\newcommand*{\shuffle}{%
    \LWR@shuffle@start%
    \LWR@shuffle@finish%
```

```
}

\newcommand*{\cshuffle}{%
    \LWR@shuffle@start%
    \put(.05,.65){\line(1,0){.65}}%
    \LWR@shuffle@finish%
}


\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\shuffle}{\mathbin{\unicode{0x29E2}}}}
\CustomizeMathJax{\newcommand{\cshuffle}{%
    \mathbin{\LWRoverlaysymbols{\raise{.6ex}{-}}{\unicode{0x29E2}}}%
}}
\end{warpMathJax}
```

File 448 **lwarp-sidecap.sty**

§ 557 Package

sidecap

(*Emulates or patches code by* Rolf Niepraschk, Hubert Gässlein.)

sidecap (*Pkg*) sidecap is emulated.

for HTML output: Discard all options for lwarp-sidecap.

```
\LWR@ProvidesPackageDrop{sidecap}[2003/06/06]
```

See:
http://tex.stackexchange.com/questions/45401/use-the-s-star-argument-with-newdocumentenvironment
regarding the creation of starred environments with xparse.

```
\NewDocumentEnvironment{SCtable}{soo}
{\IfValueTF{#3}{\table[#3]}{\table}}
{\endtable}

\ExplSyntaxOn
\cs_new:cpn {SCtable*} {\SCtable*}
\cs_new_eq:cN {endSCtable*} \endSCtable
\ExplSyntaxOff


\NewDocumentEnvironment{SCfigure}{soo}
{\IfValueTF{#3}{\figure[#3]}{\figure}}
{\endfigure}

\ExplSyntaxOn
\cs_new:cpn {SCfigure*} {\SCfigure*}
\cs_new_eq:cN {endSCfigure*} \endSCfigure
\ExplSyntaxOff


\newenvironment*{wide}{}{}
```

File 449 **lwarp-sidenotes.sty**

§ 558 Package

sidenotes

(*Emulates or patches code by* Andy Thomas, Oliver Schebaum.)

sidenotes (*Pkg*) Patched for lwarp.

for HTML output: Load the original package:

```
\LWR@ProvidesPackagePass{sidenotes}
```

The following patch sidenotes for use with lwarp.

An ARIA note role is not assigned since the caption is an important part of the figure.

\sidecaption * [⟨*entry*⟩] [⟨*offset*⟩] {⟨*text*⟩}

```
\RenewDocumentCommand \sidecaption {s o o m}
{
    \LWR@stoppars
    \begingroup
  \captionsetup{style=sidecaption}%
  \IfBooleanTF{#1}
  { % starred
    \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}%
    \caption*{#4}%
    \end{BlockClass}
  }
  { % unstarred
  \IfNoValueOrEmptyTF{#2}
    {\def\@sidenotes@sidecaption@tof{#4}}
    {\def\@sidenotes@sidecaption@tof{#2}}
    \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}%
    \caption[\@sidenotes@sidecaption@tof]{#4}
    \end{BlockClass}
  }
    \endgroup
    \LWR@startpars
}
```

Borrowed from the lwarp version of keyfloat:

```
\NewDocumentEnvironment{KFLTsidenotes@marginfloat}{O{-1.2ex} m}
{% start
    \LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}{marginblock}%
    \renewcommand*{\@captype}{#2}%
}
{%
    \endLWR@BlockClassWP%
}

\RenewDocumentEnvironment{marginfigure}{o}
  {\begin{KFLTsidenotes@marginfloat}{figure}}
  {\end{KFLTsidenotes@marginfloat}}
```

```
\RenewDocumentEnvironment{margintable}{o}
  {\begin{KFLTsidenotes@marginfloat}{table}}
  {\end{KFLTsidenotes@marginfloat}}
```

The following were changed by sidenotes, and now are reset back to their lwarp-supported originals:

Restoring the definition from the LaTeX 2ε `article.cls` source:

```
\renewenvironment{figure*}
               {\@dblfloat{figure}}
               {\end@dblfloat}

\renewenvironment{table*}
               {\@dblfloat{table}}
               {\end@dblfloat}
```

For MathJax:

⚠ Note that sidenotes does not support `\sidenote` inside math in print mode. Use `\sidenotemark` and `\sidenotetext` instead.

```
\begin{warpMathJax}
\providecommand{\sidenotename}{sidenote}
\appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRsidenote}{\thesidenote}}
\appto\LWR@syncnotenames{\LWR@synconenotename{LWRsidenote}{\sidenotename}}
\CustomizeMathJax{\def\LWRsidenote{1}}
\CustomizeMathJax{\newcommand{\sidenotemark}[1][\LWRsidenote]{{}^{\mathrm{#1}}}}
\end{warpMathJax}
```

The following is not defined since is not allowed inside math in print mode, and also would have to be modified to parse the optional offset argument:

```
\CustomizeMathJax{\newcommand{\sidenote}[2][\LWRsidenote]{{}^{\mathrm{#1}}}}
```

File 450 **`lwarp-simplebnf.sty`**

§ 559 Package

simplebnf

(*Emulates or patches code by* Jay Lee.)

`simplebnf` (*Pkg*) simplebnf is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{simplebnf}[2022/05/08]
```

The entire object is placed inside a `lateximage` whose `alt` text is the LaTeX source BNF expression.

```
\ExplSyntaxOn

\RenewDocumentEnvironment { bnfgrammar } { O{llcll} O{[^\|]\|[^\|]} O{\|\|} +b }
  {
    \regex_gset:Nn \g_simplebnf_rhs_newline_r { #2 }
    \regex_gset:Nn \g_simplebnf_rhs_nb_r { #3 }
```

```
8
9     %% \l__input_seq is a list of term definitions.
10    \regex_split:nnN { ;; } { #4 } \l__input_seq
11    \begin{center}
12    \begin{lateximage}[#4]%     lwarp
13      \tl_set:Nn \l__table_tl
14        {
15          \begin{tabular}{#1}
16        }
17
18  \bool_set_true:N \l_tmp_first_term % Is this the first term in this grammar?
19    \seq_map_inline:Nn \l__input_seq
20      {
21        %% If not-first, add newline
22        \bool_if:NTF \l_tmp_first_term
23          {
24            \bool_set_false:N \l_tmp_first_term
25          }
26          {
27            \tl_put_right:Nn \l__table_tl { \\ }
28          }
29
30        \regex_split:nnNTF { ::= } { ##1 } \l__term_seq
31          % Parse a ::= definition
32          {
33            %% \l__term_seq    - (lhs, rhses)...
34            %% \l__term_tl     - lhs
35            %% \l__keypairs_tl - rhses
36            \seq_pop_left:NN \l__term_seq \l__term_tl
37            \seq_pop_left:NN \l__term_seq \l__keypairs_tl
38
39            \simplebnf_typeset_lhs:n{\l__term_tl}
40            \tl_put_right:Nn \l__table_tl
41              {
42                & \SimpleBNFDefEq &
43              }
44            %% \l__keypairs_seq - (rhs:annot | rhs)...
45        \regex_split:NVN \g_simplebnf_rhs_newline_r \l__keypairs_tl \l__keypairs_seq
46
47            \bool_set_true:N \l__first_rhs
48            \seq_map_function:NN \l__keypairs_seq \simplebnf_typeset_rhs:n
49          }
50          {
51            % Else, parse a \in declaration
52            \regex_split:nnNTF { \c{in} } { ##1 } \l__term_seq
53              {
54                %% \l__term_seq - (lhs, rhs)
55                \seq_pop_left:NN \l__term_seq \l_tmpa_tl
56
57                \simplebnf_typeset_lhs:n{\l_tmpa_tl}
58                \tl_put_right:Nn \l__table_tl
59                  {
60                    & $\in$ & $
61                  }
62                \seq_pop_left:NN \l__term_seq \l_tmpa_tl
63                \tl_put_right:NV \l__table_tl \l_tmpa_tl
64                \tl_put_right:Nn \l__table_tl
65                  {
66                    $ &
67                  }
```

```
            }
            { \msg_error:nn {simplebnf} { Could not parser ##1 } }
        }
    }

  \tl_put_right:Nn \l__table_tl { \end{tabular} }
  \tl_use:N \l__table_tl
  \end{lateximage}%       lwarp
  \end{center}
}
{ }

\ExplSyntaxOff
```

File 451 **lwarp-SIunits.sty**

§ 560 Package

SIunits

(*Emulates or patches code by* Marcel Heldoorn.)

SIunits (*Pkg*) SIunits is patched for use by lwarp.

For SVG math, it is recommended to use `\unit` where possible, which combines the entire expression into a single `lateximage`, and adds the `alt` tag containing the LaTeX code, allowing for copy/paste. When units are used outside of the `\unit` macro, each unit macro will have its own `lateximage`, and each will have the `alt` tag set according to `\MathImageAltText`, which defaults to `(math image)`.

For MathJax, individual units used in text will appear as SVG images, since `\ensuremath` is used in the original defintions, and `\ensuremath` often has expressions which do not work well in MathJax, so it is always forced to an SVG image. If, however, `\unit` is used, the result is expressed with MathJaxinstead of an SVG image.

for HTML output:

```
\LWR@ProvidesPackagePass{SIunits}[2007/12/02]
```

Patched for copy/paste with the HTML `alt` tag:

```
\ifbool{mathjax}{
    \DeclareRobustCommand{\LWR@HTML@unit}[2]{%
        \LWR@subsingledollar*%  lwarp
        {% alt tag
            \textbackslash{}unit%
            \{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
             \{ \LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}% extra space
        }%
        {SIunits}% add'l hashing
        {%
            #1\,{#2}%
        }% contents
    }
}{% not MathJax
    \DeclareRobustCommand{\LWR@HTML@unit}[2]{%
        \@inunitcommandtrue%    original
        \LWR@subsingledollar*%  lwarp
        {% alt tag
        \textbackslash{}unit\{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
```

```
             \{ \LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}% extra space
        }%
        {SIunits}% add'l hashing
        {%
            \LWR@origensuredmath{%  lwarp modification
                \SI@fstyle{%
                    {#1}\@qsk\period@active{#2}%
                }%  original
            }%
        }% contents
        \@inunitcommandfalse%   original
    }
}% not MathJax
\LWR@formatted{unit}
```

For MathJax:

```
\begin{warpMathJax}
\LWR@infoprocessingmathjax{SIunits}

\CustomizeMathJax{\newcommand{\one}{}}
\CustomizeMathJax{\newcommand{\meter}{\metre}}
\CustomizeMathJax{\newcommand{\deka}{\deca}}
\CustomizeMathJax{\newcommand{\dekad}{\decad}}
\CustomizeMathJax{\newcommand{\per}{/}}
\CustomizeMathJax{\newcommand{\usk}{\;}}
\CustomizeMathJax{\newcommand{\unit}[2]{#1\,{#2}}}
\CustomizeMathJax{\newcommand{\power}[2]{#1^{#2}}}

\AtBeginDocument{%
 \if@redefsquare
  \CustomizeMathJax{\renewcommand{\square}[1]{\power{#1}{2}}}
 \else
   \if@defsquaren
    \CustomizeMathJax{\newcommand{\squaren}[1]{\power{#1}{2}}}
   \else
     \CustomizeMathJax{\renewcommand{\square}[1]{\power{#1}{2}}}
   \fi %\if@defsquaren
 \fi   %\if@redefsquare
}     %\AtBeginDocument

\CustomizeMathJax{\newcommand{\squared}{^{2}}}
\CustomizeMathJax{\newcommand{\cubic}[1]{\power{#1}{3}}}
\CustomizeMathJax{\newcommand{\cubed}{^{3}}}
\CustomizeMathJax{\newcommand{\fourth}[1]{\power{#1}{4}}}
\CustomizeMathJax{\newcommand{\reciprocal}[1]{\power{#1}{-1}}}
\CustomizeMathJax{\newcommand{\rp}{\reciprocal}}
\CustomizeMathJax{\newcommand{\rpsquare}[1]{\power{#1}{-2}}}
\CustomizeMathJax{\newcommand{\rpsquared}{^{-2}}}
\CustomizeMathJax{\newcommand{\rpcubic}[1]{\power{#1}{-3}}}
\CustomizeMathJax{\newcommand{\rpcubed}{^{-3}}}
\CustomizeMathJax{\newcommand{\rpfourth}[1]{\power{#1}{-4}}}
\CustomizeMathJax{\newcommand{\yocto}{\mathrm{y}}}
\CustomizeMathJax{\newcommand{\zepto}{\mathrm{z}}}
\CustomizeMathJax{\newcommand{\atto}{\mathrm{a}}}
\CustomizeMathJax{\newcommand{\femto}{\mathrm{f}}}
\CustomizeMathJax{\newcommand{\pico}{\mathrm{p}}}
\CustomizeMathJax{\newcommand{\nano}{\mathrm{n}}}
\CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
\CustomizeMathJax{\newcommand{\milli}{\mathrm{m}}}
```

```
\CustomizeMathJax{\newcommand{\centi}{\mathrm{c}}}
\CustomizeMathJax{\newcommand{\deci}{\mathrm{d}}}
\CustomizeMathJax{\newcommand{\deca}{\mathrm{da}}}
\CustomizeMathJax{\newcommand{\hecto}{\mathrm{h}}}
\CustomizeMathJax{\newcommand{\kilo}{\mathrm{k}}}
\CustomizeMathJax{\newcommand{\mega}{\mathrm{M}}}
\CustomizeMathJax{\newcommand{\giga}{\mathrm{G}}}
\CustomizeMathJax{\newcommand{\tera}{\mathrm{T}}}
\CustomizeMathJax{\newcommand{\peta}{\mathrm{P}}}
\CustomizeMathJax{\newcommand{\exa}{\mathrm{E}}}
\CustomizeMathJax{\newcommand{\zetta}{\mathrm{Z}}}
\CustomizeMathJax{\newcommand{\yotta}{\mathrm{Y}}}
\CustomizeMathJax{\newcommand{\yoctod}{\power{10}{-24}}}
\CustomizeMathJax{\newcommand{\zeptod}{\power{10}{-21}}}
\CustomizeMathJax{\newcommand{\attod}{\power{10}{-18}}}
\CustomizeMathJax{\newcommand{\femtod}{\power{10}{-15}}}
\CustomizeMathJax{\newcommand{\picod}{\power{10}{-12}}}
\CustomizeMathJax{\newcommand{\nanod}{\power{10}{-9}}}
\CustomizeMathJax{\newcommand{\microd}{\power{10}{-6}}}
\CustomizeMathJax{\newcommand{\millid}{\power{10}{-3}}}
\CustomizeMathJax{\newcommand{\centid}{\power{10}{-2}}}
\CustomizeMathJax{\newcommand{\decid}{\power{10}{-1}}}
\CustomizeMathJax{\newcommand{\decad}{\power{10}{1}}}
\CustomizeMathJax{\newcommand{\hectod}{\power{10}{2}}}
\CustomizeMathJax{\newcommand{\kilod}{\power{10}{3}}}
\CustomizeMathJax{\newcommand{\megad}{\power{10}{6}}}
\CustomizeMathJax{\newcommand{\gigad}{\power{10}{9}}}
\CustomizeMathJax{\newcommand{\terad}{\power{10}{12}}}
\CustomizeMathJax{\newcommand{\petad}{\power{10}{15}}}
\CustomizeMathJax{\newcommand{\exad}{\power{10}{18}}}
\CustomizeMathJax{\newcommand{\zettad}{\power{10}{21}}}
\CustomizeMathJax{\newcommand{\yottad}{\power{10}{24}}}
\CustomizeMathJax{\newcommand{\gram}{\mathrm{g}}}
\CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\kilogram}{\kilo\gram}}
\CustomizeMathJax{\newcommand{\second}{\mathrm{s}}}
\CustomizeMathJax{\newcommand{\ampere}{\mathrm{A}}}
\CustomizeMathJax{\newcommand{\kelvin}{\mathrm{K}}}
\CustomizeMathJax{\newcommand{\mole}{\mathrm{mol}}}
\CustomizeMathJax{\newcommand{\candela}{\mathrm{cd}}}
\CustomizeMathJax{\newcommand{\radian}{\mathrm{rad}}}
\CustomizeMathJax{\newcommand{\steradian}{\mathrm{sr}}}
\CustomizeMathJax{\newcommand{\hertz}{\mathrm{Hz}}}
\CustomizeMathJax{\newcommand{\newton}{\mathrm{N}}}
\CustomizeMathJax{\newcommand{\pascal}{\mathrm{Pa}}}
\CustomizeMathJax{\newcommand{\joule}{\mathrm{J}}}
\CustomizeMathJax{\newcommand{\watt}{\mathrm{W}}}
\CustomizeMathJax{\newcommand{\coulomb}{\mathrm{C}}}
\CustomizeMathJax{\newcommand{\volt}{\mathrm{V}}}
\CustomizeMathJax{\newcommand{\farad}{\mathrm{F}}}
\CustomizeMathJax{\newcommand{\ohm}{\mathrm{\Omega}}}
\CustomizeMathJax{\newcommand{\siemens}{\mathrm{S}}}
\CustomizeMathJax{\newcommand{\weber}{\mathrm{Wb}}}
\CustomizeMathJax{\newcommand{\tesla}{\mathrm{T}}}
\CustomizeMathJax{\newcommand{\henry}{\mathrm{H}}}
\CustomizeMathJax{\newcommand{\degreecelsius}{\mathrm{\unicode{x2103}}}}
\CustomizeMathJax{\newcommand{\celsius}{\degreecelsius}}
\CustomizeMathJax{\newcommand{\lumen}{\mathrm{lm}}}
\CustomizeMathJax{\newcommand{\lux}{\mathrm{lx}}}
\CustomizeMathJax{\newcommand{\becquerel}{\mathrm{Bq}}}
```

```
\CustomizeMathJax{\newcommand{\sievert}{\mathrm{Sv}}}
\CustomizeMathJax{\newcommand{\katal}{\mathrm{kat}}}

\ifdef{\radianbase}{
\CustomizeMathJax{\newcommand{\radianbase}%
        {\metre\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\steradianbase}%
        {\squaremetre\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\hertzbase}%
        {\reciprocal\second}}
\CustomizeMathJax{\newcommand{\newtonbase}%
        {\metre\usk\kilogram\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\pascalbase}%
        {\reciprocal\metre\usk\kilogram\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\joulebase}%
        {\squaremetre\usk\kilogram\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\wattbase}%
        {\squaremetre\usk\kilogram\usk\rpcubic\second}}
\CustomizeMathJax{\newcommand{\coulombbase}%
        {\ampere\usk\second}}
\CustomizeMathJax{\newcommand{\voltbase}%
       {\squaremetre\usk\kilogram\usk\rpcubic\second\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\faradbase}%
     {\rpsquare\metre\usk\reciprocal\kilogram\usk\fourth\second\usk\ampere\squared}}
\CustomizeMathJax{\newcommand{\ohmbase}%
        {\squaremetre\usk\kilogram\usk\rpcubic\second\usk\rpsquare\ampere}}
\CustomizeMathJax{\newcommand{\siemensbase}%
     {\rpsquare\metre\usk\reciprocal\kilogram\usk\cubic\second\usk\ampere\squared}}
\CustomizeMathJax{\newcommand{\weberbase}%
     {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\teslabase}%
        {\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\henrybase}%
       {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\rpsquare\ampere}}
\CustomizeMathJax{\newcommand{\celsiusbase}%
        {\kelvin}}
\CustomizeMathJax{\newcommand{\lumenbase}%
        {\candela\usk\squaremetre\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\luxbase}%
        {\candela\usk\squaremetre\usk\rpfourth\metre}}
\CustomizeMathJax{\newcommand{\becquerelbase}%
        {\hertzbase}}
\CustomizeMathJax{\newcommand{\graybase}%
        {\squaremetre\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\sievertbase}%
        {\graybase}}
\CustomizeMathJax{\newcommand{\katalbase}%
        {\rp\second\usk\mole }}
}{}

\ifdef{\derradian}{
\CustomizeMathJax{\newcommand{\derradian}%
        {\metre\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\dersteradian}%
        {\squaremetre\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\derhertz}%
        {\reciprocal\second}}
\CustomizeMathJax{\newcommand{\dernewton}%
        {\metre\usk\kilogram\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\derpascal}%
```

```
        {\newton\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\derjoule}%
        {\newton\usk\metre}}
\CustomizeMathJax{\newcommand{\derwatt}%
        {\joule\usk\reciprocal\second}}
\CustomizeMathJax{\newcommand{\dercoulomb}%
        {\ampere\usk\second}}
\CustomizeMathJax{\newcommand{\dervolt}%
        {\watt\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\derfarad}%
        {\coulomb\usk\reciprocal\volt}}
\CustomizeMathJax{\newcommand{\derohm}%
        {\volt\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\dersiemens}%
        {\ampere\usk\reciprocal\volt}}
\CustomizeMathJax{\newcommand{\derweber}%
     {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\dertesla}%
        {\weber\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\derhenry}%
        {\weber\usk\reciprocal\ampere}}
\CustomizeMathJax{\newcommand{\dercelsius}%
        {\kelvin}}
\CustomizeMathJax{\newcommand{\derlumen}%
        {\candela\usk\steradian}}
\CustomizeMathJax{\newcommand{\derlux}%
        {\lumen\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\derbecquerel}%
        {\derhertz}}
\CustomizeMathJax{\newcommand{\dergray}%
        {\joule\usk\reciprocal\kilogram}}
\CustomizeMathJax{\newcommand{\dersievert}%
        {\dergray}}
\CustomizeMathJax{\newcommand{\derkatal}%
        {\katalbase}}
}{}

\CustomizeMathJax{\newcommand{\minute}{\mathrm{min}}}
\CustomizeMathJax{\newcommand{\hour}{\mathrm{h}}}
\CustomizeMathJax{\newcommand{\dday}{\mathrm{d}}}
\CustomizeMathJax{\newcommand{\degree}{\mathrm{^\circ}}}
\CustomizeMathJax{\newcommand{\paminute}{^\prime}}
\CustomizeMathJax{\newcommand{\arcminute}{^\prime}}
\CustomizeMathJax{\newcommand{\pasecond}{^{\prime\prime}}}
\CustomizeMathJax{\newcommand{\arcsecond}{^{\prime\prime}}}
\CustomizeMathJax{\newcommand{\ton}{\mathrm{t}}}
\CustomizeMathJax{\newcommand{\tonne}{\mathrm{t}}}
\CustomizeMathJax{\newcommand{\liter}{\mathrm{L}}}
\CustomizeMathJax{\newcommand{\litre}{\mathrm{l}}}
\CustomizeMathJax{\newcommand{\neper}{\mathrm{Np}}}
\CustomizeMathJax{\newcommand{\bel}{\mathrm{B}}}
\CustomizeMathJax{\newcommand{\curie}{\mathrm{Ci}}}
\CustomizeMathJax{\newcommand{\rad}{\mathrm{rad}}}
\CustomizeMathJax{\newcommand{\arad}{\mathrm{rd}}}
\CustomizeMathJax{\newcommand{\rem}{\mathrm{rem}}}
\CustomizeMathJax{\newcommand{\roentgen}{\mathrm{R}}}
\CustomizeMathJax{\newcommand{\electronvolt}{\mathrm{\mathrm{eV}}}}
\CustomizeMathJax{\newcommand{\atomicmass}{\mathrm{u}}}
\CustomizeMathJax{\newcommand{\atomicmassunit}{\mathrm{u}}}
\CustomizeMathJax{\newcommand{\dalton}{\mathrm{Da}}}
```

```
\CustomizeMathJax{\newcommand{\are}{\mathrm{a}}}
\CustomizeMathJax{\newcommand{\hectare}{\mathrm{\hecto\are}}}
\CustomizeMathJax{\newcommand{\barn}{\mathrm{b}}}
\CustomizeMathJax{\newcommand{\bbar}{\mathrm{bar}}}
\CustomizeMathJax{\newcommand{\gal}{\mathrm{Gal}}}
\CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
\CustomizeMathJax{\newcommand{\rperminute}{\mathrm{r}\per\minute}}
\CustomizeMathJax{\newcommand{\rpersecond}{\mathrm{r}\per\second}}
\CustomizeMathJax{\newcommand{\squaremetre}{\power{\metre}{2}}}
\CustomizeMathJax{\newcommand{\cubicmetre}{\cubic\metre}}
\CustomizeMathJax{\newcommand{\graypersecond}{\gray\per\second}}
\CustomizeMathJax{\newcommand{\graypersecondnp}{\gray\usk\reciprocal\second}}
\CustomizeMathJax{\newcommand{\metrepersquaresecond}{\metre\per\second\squared}}
\CustomizeMathJax{\newcommand{\metrepersquaresecondnp}{\metre\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\joulepermole}{\joule\per\mole}}
\CustomizeMathJax{\newcommand{\joulepermolenp}{\joule\usk\reciprocal\mole}}
\CustomizeMathJax{\newcommand{\molepercubicmetre}{\mole\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\molepercubicmetrenp}{\mole\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\radianpersquaresecond}{\radian\per\second\squared}}
\CustomizeMathJax{\newcommand{\radianpersquaresecondnp}{\radian\usk\second\rpsquared}}
\CustomizeMathJax{\newcommand{\kilogramsquaremetrepersecond}{%
    \kilogram\usk\squaremetre\per\second%
}}
\CustomizeMathJax{\newcommand{\kilogramsquaremetrepersecondnp}{%
    \kilogram\usk\squaremetre\usk\reciprocal\second%
}}
\CustomizeMathJax{\newcommand{\radianpersecond}{\radian\per\second}}
\CustomizeMathJax{\newcommand{\radianpersecondnp}{\radian\usk\reciprocal\second}}
\CustomizeMathJax{\newcommand{\squaremetrepercubicmetre}{\squaremetre\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\squaremetrepercubicmetrenp}{%
    \squaremetre\usk\rpcubic\metre%
}}
\CustomizeMathJax{\newcommand{\katalpercubicmetre}{\katal\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\katalpercubicmetrenp}{\katal\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\coulombpermol}{\coulomb\per\mole}}
\CustomizeMathJax{\newcommand{\coulombpermolnp}{\coulomb\usk\reciprocal\mole}}
\CustomizeMathJax{\newcommand{\amperepersquaremetre}{\ampere\per\squaremetre}}
\CustomizeMathJax{\newcommand{\amperepersquaremetrenp}{\ampere\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\kilogrampercubicmetre}{\kilogram\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\kilogrampercubicmetrenp}{\kilogram\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\squaremetrepernewtonsecond}{%
    \squaremetre\per\newton\usk\second%
}}
\CustomizeMathJax{\newcommand{\squaremetrepernewtonsecondnp}{%
    \squaremetre\usk\reciprocal\newton\usk\reciprocal\second%
}}
\CustomizeMathJax{\newcommand{\pascalsecond}{\pascal\usk\second}}
\CustomizeMathJax{\newcommand{\coulombpercubicmetre}{\coulomb\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\coulombpercubicmetrenp}{\coulomb\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\amperemetresecond}{\ampere\usk\metre\usk\second}}
\CustomizeMathJax{\newcommand{\voltpermetre}{\volt\per\metre}}
\CustomizeMathJax{\newcommand{\voltpermetrenp}{\volt\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\coulombpersquaremetre}{\coulomb\per\squaremetre}}
\CustomizeMathJax{\newcommand{\coulombpersquaremetrenp}{\coulomb\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\faradpermetre}{\farad\per\metre}}
\CustomizeMathJax{\newcommand{\faradpermetrenp}{\farad\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\ohmmetre}{\ohm\usk\metre}}
\CustomizeMathJax{\newcommand{\kilowatthour}{\kilo\watt\hour}}
\CustomizeMathJax{\newcommand{\wattpersquaremetre}{\watt\per\squaremetre}}
\CustomizeMathJax{\newcommand{\wattpersquaremetrenp}{\watt\usk\rpsquare\metre}}
```

```
\CustomizeMathJax{\newcommand{\joulepersquaremetre}{\joule\per\squaremetre}}
\CustomizeMathJax{\newcommand{\joulepersquaremetrenp}{\joule\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\newtonpercubicmetre}{\newton\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\newtonpercubicmetrenp}{\newton\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\newtonperkilogram}{\newton\per\kilogram}}
\CustomizeMathJax{\newcommand{\newtonperkilogramnp}{\newton\usk\reciprocal\kilogram}}
\CustomizeMathJax{\newcommand{\jouleperkelvin}{\joule\per\kelvin}}
\CustomizeMathJax{\newcommand{\jouleperkelvinnp}{\joule\usk\reciprocal\kelvin}}
\CustomizeMathJax{\newcommand{\jouleperkilogram}{\joule\per\kilogram}}
\CustomizeMathJax{\newcommand{\jouleperkilogramnp}{\joule\usk\reciprocal\kilogram}}
\CustomizeMathJax{\newcommand{\coulombperkilogram}{\coulomb\per\kilogram}}
\CustomizeMathJax{\newcommand{\coulombperkilogramnp}{\coulomb\usk\reciprocal\kilogram}}
\CustomizeMathJax{\newcommand{\squaremetrepersecond}{\squaremetre\per\second}}
\CustomizeMathJax{\newcommand{\squaremetrepersecondnp}{%
    \squaremetre\usk\reciprocal\second%
}}
\CustomizeMathJax{\newcommand{\squaremetrepersquaresecond}{%
    \squaremetre\per\second\squared%
}}
\CustomizeMathJax{\newcommand{\squaremetrepersquaresecondnp}{%
    \squaremetre\usk\second\rpsquared%
}}
\CustomizeMathJax{\newcommand{\kilogrammetrepersecond}{%
    \kilogram\usk\metre\per\second%
}}
\CustomizeMathJax{\newcommand{\kilogrammetrepersecondnp}{%
    \kilogram\usk\metre\usk\reciprocal\second%
}}
\CustomizeMathJax{\newcommand{\candelapersquaremetre}{\candela\per\squaremetre}}
\CustomizeMathJax{\newcommand{\candelapersquaremetrenp}{\candela\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\amperepermetre}{\ampere\per\metre}}
\CustomizeMathJax{\newcommand{\amperepermetrenp}{\ampere\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\joulepertesla}{\joule\per\tesla}}
\CustomizeMathJax{\newcommand{\joulepertesla np}{\joule\usk\reciprocal\tesla}}
\CustomizeMathJax{\newcommand{\henrypermetre}{\henry\per\metre}}
\CustomizeMathJax{\newcommand{\henrypermetrenp}{\henry\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\kilogrampersecond}{\kilogram\per\second}}
\CustomizeMathJax{\newcommand{\kilogrampersecondnp}{\kilogram\usk\reciprocal\second}}
\CustomizeMathJax{\newcommand{\kilogrampersquaremetresecond}{%
    \kilogram\per\squaremetre\usk\second%
}}
\CustomizeMathJax{\newcommand{\kilogrampersquaremetresecondnp}{%
    \kilogram\usk\rpsquare\metre\usk\reciprocal\second%
}}
\CustomizeMathJax{\newcommand{\kilogrampersquaremetre}{\kilogram\per\squaremetre}}
\CustomizeMathJax{\newcommand{\kilogrampersquaremetrenp}{\kilogram\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\kilogrampermetre}{\kilogram\per\metre}}
\CustomizeMathJax{\newcommand{\kilogrampermetrenp}{\kilogram\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\joulepermolekelvin}{\joule\per\mole\usk\kelvin}}
\CustomizeMathJax{\newcommand{\joulepermolekelvinnp}{%
    \joule\usk\reciprocal\mole\usk\reciprocal\kelvin%
}}
\CustomizeMathJax{\newcommand{\kilogramperkilomole}{\kilogram\per\kilo\mole}}
\CustomizeMathJax{\newcommand{\kilogramperkilomolenp}{%
    \kilogram\usk\kilo\reciprocal\mole%
}}
\CustomizeMathJax{\newcommand{\kilogramsquaremetre}{\kilogram\usk\squaremetre}}
\CustomizeMathJax{\newcommand{\kilogramsquaremetrenp}{\kilogramsquaremetre}}
\CustomizeMathJax{\newcommand{\kilogrammetrepersquaresecond}{%
    \kilogram\usk\metre\per\second\squared%
```

```
}}
\CustomizeMathJax{\newcommand{\kilogrammetrepersquaresecondnp}{%
    \kilogram\usk\metre\usk\second\rpsquared%
}}
\CustomizeMathJax{\newcommand{\newtonpersquaremetre}{\newton\per\squaremetre}}
\CustomizeMathJax{\newcommand{\newtonpersquaremetrenp}{\newton\usk\rpsquare\metre}}
\CustomizeMathJax{\newcommand{\persquaremetresecond}{1\per\squaremetre\usk\second}}
\CustomizeMathJax{\newcommand{\persquaremetresecondnp}{%
    \rpsquare\metre\usk\reciprocal\second%
}}
\CustomizeMathJax{\newcommand{\wattperkilogram}{\watt\per\kilogram}}
\CustomizeMathJax{\newcommand{\wattperkilogramnp}{\watt\usk\reciprocal\kilogram}}
\CustomizeMathJax{\newcommand{\wattpercubicmetre}{\watt\per\cubic\metre}}
\CustomizeMathJax{\newcommand{\wattpercubicmetrenp}{\watt\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\wattpersquaremetresteradian}{%
    \watt\per\squaremetre\usk\steradian%
}}
\CustomizeMathJax{\newcommand{\wattpersquaremetresteradiannp}{%
    \watt\usk\rpsquare\metre\usk\rp\steradian%
}}
\CustomizeMathJax{\newcommand{\jouleperkilogramkelvin}{\joule\per\kilogram\usk\kelvin}}
\CustomizeMathJax{\newcommand{\jouleperkilogramkelvinnp}{%
    \joule\usk\reciprocal\kilogram\usk\reciprocal\kelvin%
}}
\CustomizeMathJax{\newcommand{\squaremetreperkilogram}{\squaremetre\per\kilogram}}
\CustomizeMathJax{\newcommand{\rpsquaremetreperkilogram}{%
    \squaremetre\usk\reciprocal\kilogram%
}}
\CustomizeMathJax{\newcommand{\cubicmetreperkilogram}{\cubic\metre\per\kilogram}}
\CustomizeMathJax{\newcommand{\rpcubicmetreperkilogram}{%
    \cubic\metre\usk\reciprocal\kilogram%
}}
\CustomizeMathJax{\newcommand{\newtonpermetre}{\newton\per\metre}}
\CustomizeMathJax{\newcommand{\newtonpermetrenp}{\newton\usk\reciprocal\metre}}
\CustomizeMathJax{\newcommand{\Celsius}{\unicode{x2103}}}
\CustomizeMathJax{\newcommand{\wattpermetrekelvin}{\watt\per\metre\usk\kelvin}}
\CustomizeMathJax{\newcommand{\wattpermetrekelvinnp}{%
    \watt\usk\reciprocal\metre\usk\reciprocal\kelvin%
}}
\CustomizeMathJax{\newcommand{\newtonmetre}{\newton\usk\metre}
\CustomizeMathJax{\newcommand{\newtonmetrenp}{\newtonmetre}}}
\CustomizeMathJax{\newcommand{\squaremetrepercubicsecond}{%
    \squaremetre\per\cubic\second%
}}
\CustomizeMathJax{\newcommand{\squaremetrepercubicsecondnp}{%
    \squaremetre\usk\rpcubic\second%
}}
\CustomizeMathJax{\newcommand{\metrepersecond}{\metre\per\second}}
\CustomizeMathJax{\newcommand{\metrepersecondnp}{\metre\usk\reciprocal\second}}
\CustomizeMathJax{\newcommand{\joulepercubicmetre}{\joule\per\cubicmetre}}
\CustomizeMathJax{\newcommand{\joulepercubicmetrenp}{\joule\usk\rpcubic\metre}}
\CustomizeMathJax{\newcommand{\kilogrampercubicmetrecoulomb}{%
    \kilogram\per\cubic\metre\usk\coulomb%
}}
\CustomizeMathJax{\newcommand{\kilogrampercubicmetrecoulombnp}{%
    \kilogram\usk\rpcubic\metre\usk\reciprocal\coulomb%
}}
\CustomizeMathJax{\newcommand{\cubicmetrepersecond}{\cubicmetre\per\second}}
\CustomizeMathJax{\newcommand{\rpcubicmetrepersecond}{\cubicmetre\usk\reciprocal\second}}
\CustomizeMathJax{\newcommand{\kilogrampersecondcubicmetre}{%
```

```
    \kilogram\per\second\usk\cubicmetre%
}}
\CustomizeMathJax{\newcommand{\kilogrampersecondcubicmetrenp}{%
    \kilogram\usk\reciprocal\second\usk\rpcubic\metre%
}}
\end{warpMathJax}
```

File 452 **lwarp-siunitx.sty**

§ 561 Package siunitx

(*Emulates or patches code by* Joseph Wright.)

siunitx (*Pkg*) siunitx is patched for use by lwarp, and is emulated for MathJax.

for HTML output:

```
\providecommand\DeclareRelease[3]{}
\providecommand\DeclareCurrentRelease[2]{}

\DeclareRelease{2}{2010-05-23}{lwarp-siunitx-v2.sty}
\DeclareRelease{v2}{2010-05-23}{lwarp-siunitx-v2.sty}
\DeclareCurrentRelease{}{2021-05-17}

\RequirePackage{xcolor}% for \convertcolorspec

\LWR@ProvidesPackagePass{siunitx}[2022-02-15]

\ExplSyntaxOn

\cs_set_protected:Npn \siunitx_number_format:nN #1#2
  {
    \group_begin:
      \bool_if:NTF \l_siunitx_number_parse_bool
        {
          \siunitx_number_parse:nN {#1} \l__siunitx_number_parsed_tl
       \siunitx_number_process:NN \l__siunitx_number_parsed_tl \l__siunitx_number_parsed_tl
          \tl_set:Nx \l__siunitx_number_outputted_tl
            { \siunitx_number_output:N \l__siunitx_number_parsed_tl }
        }
        {
            \tl_set:Nn \l__siunitx_number_outputted_tl
            {
                \LWR@subsingledollar{%                  lwarp
                    \textbackslash( % space
                    \LWR@HTMLsanitizedetokenized{%
                        \detokenize{#1}%
                    } \textbackslash)%                  lwarp
                }%
                {siunitx unparsed}%
                {\ensuremath{#1}}%                      lwarp
            }
        }
    \exp_args:NNNV \group_end:
    \tl_set:Nn #2 \l__siunitx_number_outputted_tl
  }

\cs_set_protected:Npn \__siunitx_compound_unparsed:n #1
  {
```

```
    \tl_if_blank:nF {#1}
      { \seq_put_right:Nn \l__siunitx_compound_tmp_seq
        {
            \LWR@subsingledollar{%                  lwarp
                \textbackslash( % space
                \LWR@HTMLsanitizedetokenized{%
                    \detokenize{#1}%
                } \textbackslash)%                  lwarp
            }%
            {siunitx unparsed}%
            {\ensuremath{#1}}%                      lwarp
        }
      }
  }
```

If not in a `lateximage`, always use text mode. Ignore current text font if resetting text family, series, and shape.

```
\cs_set_protected:Npn \__siunitx_print_aux:nn #1#2
  {
    \tl_if_empty:cTF { l__siunitx_print_ #1 _color_tl }
      { \use:n }
      { \exp_args:Nv \textcolor { l__siunitx_print_ #1 _color_tl } }
        {
            \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%      lwarp
            {
                \use:c
                    {
                    siunitx_print_
                    \tl_use:c { l__siunitx_print_ #1 _mode_tl } :n
                    }
                    {#2}
            }
            {
                \bool_lazy_all:nTF%     lwarp
                    {
                        {\l__siunitx_print_text_family_bool}
                        {\l__siunitx_print_text_series_bool}
                        {\l__siunitx_print_text_shape_bool}
                    }
                    {% No font control if reset-text-family/series/shape
                        \use:c
                            {
                            siunitx_print_%                         lwarp
                                text%                               lwarp
                                :n%                                 lwarp
                            }%                                      lwarp
                            {#2}%                                   lwarp
                    }
                    {
                        \LWR@textcurrentfont{%                      lwarp
                            \use:c
                                {
                                siunitx_print_%                     lwarp
                                    text%                           lwarp
                                    :n%                             lwarp
                                }%                                  lwarp
                                {#2}%                               lwarp
                        }
                    }
```

```
            }
        }
}
```

To determine whether to make a complex root be italic or upright, `\l__siunitx_complex_output_root_tl` is compared to `\LWR@siunitx@complexrm<i/j>`, and the css style is set appropriately.

```
\newcommand*{\LWR@siunitx@complexrootstyle}{textrm}

\newcommand*{\LWR@siunitx@complexrmi}{\mathrm{i}}
\newcommand*{\LWR@siunitx@complexrmj}{\mathrm{j}}

\newcommand*{\LWR@siunitx@setcomplexroot}{%
    \renewcommand*{\LWR@siunitx@complexrootstyle}{textit}%
  \ifdefequal{\l__siunitx_complex_output_root_tl}{\LWR@siunitx@complexrmi}%
       {\renewcommand*{\LWR@siunitx@complexrootstyle}{textrm}}%
       {}%
  \ifdefequal{\l__siunitx_complex_output_root_tl}{\LWR@siunitx@complexrmj}%
       {\renewcommand*{\LWR@siunitx@complexrootstyle}{textrm}}%
       {}%
}

\cs_set_protected:Npn \__siunitx_complex_format_auxii:n #1
  {
    \LWR@siunitx@setcomplexroot%                          lwarp
    \__siunitx_complex_format_units:n {#1}
    \tl_if_empty:NF \l__siunitx_complex_real_tl
    { \exp_after:wN \__siunitx_complex_drop_exponent:nnnnnnn \l__siunitx_complex_real_tl }
   \exp_after:wN \__siunitx_complex_format_sign:nnnnnnn \l__siunitx_complex_img_tl
    \tl_set:Nx \l__siunitx_complex_tmp_tl
      { \siunitx_number_output:NN \l__siunitx_complex_img_tl \q_nil }
   \exp_after:wN \__siunitx_complex_extract_exponent:w \l__siunitx_complex_tmp_tl \q_stop
    \tl_set:Nx \l__siunitx_complex_tmp_tl
      {
        \bool_lazy_or:nnTF
          {
            \bool_lazy_and_p:nn
              { \l_siunitx_number_bracket_ambiguous_bool }
              { ! \tl_if_empty_p:N \l__siunitx_complex_exp_tl }
          }
          {
            ! \bool_lazy_any_p:n
              {
                { \tl_if_blank_p:n {#1} }
                { \tl_if_empty_p:N \l__siunitx_complex_real_tl }
                { \tl_if_empty_p:N \l__siunitx_complex_img_tl }
              }
          }
          { \__siunitx_complex_format_bracket:n }
          { \use:n }
          {
            \siunitx_number_output:N \l__siunitx_complex_real_tl
            \exp_not:V \l__siunitx_complex_sign_tl
            \bool_if:NF \l__siunitx_complex_root_after_bool
              {
                \InlineClass{\LWR@siunitx@complexrootstyle}%    lwarp
                    {
                        \exp_not:V \l__siunitx_complex_output_root_tl
                    }
```

```
              }
             \exp_not:V \l__siunitx_complex_tmp_tl
            \bool_if:NT \l__siunitx_complex_root_after_bool
              {
                \InlineClass{\LWR@siunitx@complexrootstyle}%    lwarp
                    {
                        \exp_not:V \l__siunitx_complex_output_root_tl
                    }
              }
           }
         \exp_not:V \l__siunitx_complex_exp_tl
      }
  }
```

{⟨*1: deg/min/sec character*⟩} {⟨*2: ?*⟩} {⟨*3: ?*⟩} {⟨*4: integer part of angle*⟩} {⟨*5: decimal point character*⟩} {⟨*6: decimal part of angle*⟩} {⟨*7: ?*⟩} {⟨*8: ?*⟩}

If not in a `lateximage`, print a simplified verison without the box measurement things which conflict with lwarp:

```
\cs_set_protected:Npn \__siunitx_angle_arc_print_auxii:nw
  #1#2 \q_nil #3 \q_nil #4 \q_nil #5 \q_nil #6 \q_nil #7 \q_nil #8 \q_stop
  {
    \mode_if_math:TF
      { \bool_set_true:N \l__siunitx_angle_tmp_bool }
      { \bool_set_false:N \l__siunitx_angle_tmp_bool }
    \siunitx_print_number:n {#2#3#4}
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%      lwarp
    {%                                                  lwarp
        \tl_if_blank:nTF {#6}
        { \__siunitx_angle_arc_print_auxvi:n {#1} }
        {
            \hbox_set:Nn \l__siunitx_angle_marker_box
            {
                \__siunitx_angle_arc_print_auxiii:n
                { \siunitx_print_number:n {#5} }
            }
            \hbox_set:Nn \l__siunitx_angle_unit_box
            {
                \__siunitx_angle_arc_print_auxiii:n
                {
                    \siunitx_unit_format:nN {#1} \l__siunitx_angle_tmp_tl
                    \siunitx_print_unit:V \l__siunitx_angle_tmp_tl
                    \skip_horizontal:n { -\scriptspace }
                }
            }
            \dim_compare:nNnTF { \box_wd:N \l__siunitx_angle_marker_box } >
            { \box_wd:N \l__siunitx_angle_unit_box }
            {
                \__siunitx_angle_arc_print_auxiv:NN
                \l__siunitx_angle_marker_box
                \l__siunitx_angle_unit_box
            }
            {
                \__siunitx_angle_arc_print_auxiv:NN
                \l__siunitx_angle_unit_box
                \l__siunitx_angle_marker_box
            }
            \hbox_set_to_wd:Nnn \l__siunitx_angle_marker_box
            \l__siunitx_angle_tmp_dim
```

```
            {
                \hbox_overlap_right:n
                { \box_use_drop:N \l__siunitx_angle_marker_box }
                \hbox_overlap_right:n
                { \box_use_drop:N \l__siunitx_angle_unit_box }
                \tex_hfil:D
            }
            \box_use:N \l__siunitx_angle_marker_box
            \skip_horizontal:N \scriptspace
            \siunitx_print_number:n {#6}
        }
    }%
```

{⟨*1: deg/min/sec character*⟩} {⟨*2: ?*⟩} {⟨*3: ?*⟩} {⟨*4: integer part of angle*⟩} {⟨*5: decimal point character*⟩} {⟨*6: decimal part of angle*⟩} {⟨*7: ?*⟩} {⟨*8: ?*⟩}

```
    {%          lwarp: not in a lateximage, simplify for HTML
        \tl_if_blank:nTF {#6}
        { \__siunitx_angle_arc_print_auxvi:n {#1} }
        {
            \__siunitx_angle_arc_print_auxiii:n
            {
                \siunitx_print_number:n {#5}
            }
            \__siunitx_angle_arc_print_auxiii:n
            {
                \siunitx_unit_format:nN {#1} \l__siunitx_angle_tmp_tl
                \siunitx_print_unit:V \l__siunitx_angle_tmp_tl
            }
            \siunitx_print_number:n {#6}
        }
    }%          lwarp
}
```

If not in a `lateximage`, print a simple inline fraction, avoiding the use of SVG math:

```
\cs_set_protected:Npn \__siunitx_print_text_fraction:Nnn #1#2#3
  {
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}% lwarp
    {%
      \ensuremath
      {
        #1
          { \mbox { \__siunitx_print_text_replace:n {#2} } }
          { \mbox { \__siunitx_print_text_replace:n {#3} } }
      }
    }%
    {%                                                          lwarp
          { \mbox { \__siunitx_print_text_replace:n {#2} } }%   lwarp
            /%                                                  lwarp
          { \mbox { \__siunitx_print_text_replace:n {#3} } }%   lwarp
    }%      lwarp
  }
```

If not in a `lateximage`, print a `\textsubscript`:

```
\cs_set_protected:Npn \__siunitx_unit_format_qualifier_subscript:
  {
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%      lwarp
```

```
    {%
        \__siunitx_unit_format_font:
        \tl_set:Nx \l__siunitx_unit_part_tl
        {
            \c__siunitx_unit_math_subscript_tl
            {
                \exp_not:V \l_siunitx_unit_font_tl
                { \exp_not:V \l__siunitx_unit_part_tl }
            }
        }
    }
    {%      lwarp simplified for HTML:
        \__siunitx_unit_format_font:
        \tl_set:Nx \l__siunitx_unit_part_tl
        {
            \textsubscript
            {
                \exp_not:V \l_siunitx_unit_font_tl
                { \exp_not:V \l__siunitx_unit_part_tl }
            }
        }
    }
  }

\cs_set_protected:Npn \siunitx_quantity:nn #1#2
  {
    \group_begin:
      \siunitx_unit_options_apply:n {#2}
      \tl_if_blank:nTF {#1}
        {
          \siunitx_unit_format:nN {#2} \l__siunitx_quantity_unit_tl
          \siunitx_print_unit:V \l__siunitx_quantity_unit_tl
        }
        {
          \bool_if:NTF \l_siunitx_number_parse_bool
            { \__siunitx_quantity_parsed:nn {#1} {#2} }
            {
              \tl_set:Nn \l__siunitx_quantity_number_tl {
                \LWR@subsingledollar{%                  lwarp
                    \textbackslash( % space
                    \LWR@HTMLsanitizedetokenized{%
                        \detokenize{#1}%
                    } \textbackslash)%                  lwarp
                }%
                {siunitx unparsed}%
                {\ensuremath{#1}}%                      lwarp
              }
              \siunitx_unit_format:nN {#2} \l__siunitx_quantity_unit_tl
              \siunitx_quantity_print:VV
                \l__siunitx_quantity_number_tl \l__siunitx_quantity_unit_tl
            }
        }
    \group_end:
  }
```

\cancel for HTML does not work yet.

```
\newcommand*{\LWR@siunitx@nocancel}[1]{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {\cancel{#1}}%  SVG
```

```
        {#1}%           HTML
}

\AtBeginDocument{
\__siunitx_unit_set_symbolic:Npnn \cancel
  { }
%   { \__siunitx_unit_parse_special:n { \cancel } }
  { \__siunitx_unit_parse_special:n { \LWR@siunitx@nocancel } }%    lwarp
}
```

For HTML, use a simple unaligned \num:

```
\newcommand{\LWR@HTML@tablenum}[2][]{\num[#1]{#2}}
\LWR@formatted{tablenum}
```

For HTML, the S column is simplified to a c column. Keys are set locally, allowing drop-exponent, etc.

```
\AtBeginDocument{
\HTMLnewcolumntype{S}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
}
```

To define simplified units for HTML:

\HTMLDeclareSIUnit [⟨*options*⟩] {⟨*name*⟩} {⟨*definition*⟩}

```
\NewDocumentCommand{\HTMLDeclareSIUnit}{o +m m}
{
    \ifcsdef{ __siunitx_unit_ \token_to_str:N #2 :w }
        {}
        {
            \PackageError{lwarp}
                {%
                    First~use\MessageBreak
                        \space\space\protect\DeclareSIUnit{
                          \token_to_str:N#2}{...}\MessageBreak
                    before~using\MessageBreak
                        \space\space\protect\HTMLDeclareSIUnit{
                          \token_to_str:N#2}{...}%
                }
                {%
                    See~the~Lwarp~manual~section~about~special~cases,~
                    regarding~siunitx.%
                }
        }
    \csNewCommandCopycs
        { __orig_siunitx_unit_ \token_to_str:N #2 :w }
        { __siunitx_unit_ \token_to_str:N #2 :w }
    \DeclareSIUnit[#1]{#2}
        {
            \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}
                {\csuse{ __orig_siunitx_unit_ \token_to_str:N #2 :w }}
                {#3}
        }
}

\ExplSyntaxOff
```

HTML versions for existing units:

```
\AtBeginDocument{
\HTMLDeclareSIUnit\celsius{\LWR@siunitx@textcelsius}
\HTMLDeclareSIUnit\arcminute{\LWR@siunitx@textprime}
\HTMLDeclareSIUnit\arcsecond{\LWR@siunitx@textdblprime}
\HTMLDeclareSIUnit\elementarycharge{\textit{e}}
%
\HTMLDeclareSIUnit\clight{\text{\textit{c}\textsubscript{0}}}
\HTMLDeclareSIUnit\bohr{\text{\textit{a}\textsubscript{0}}}
\HTMLDeclareSIUnit\electronmass{\text{\textit{m}\textsubscript{e}}}
\HTMLDeclareSIUnit\hartree{\text{\textit{E}\textsubscript{h}}}
\HTMLDeclareSIUnit\planckbar{\LWR@siunitx@textplanckbar}
}% \AtBeginDocument
```

Initial options:

```
\AtBeginDocument{
\sisetup{
    per-mode=symbol,    % fraction is not seen by pdftotext
    angle-symbol-degree = {\LWR@siunitx@textdegree},
    angle-symbol-minute = {\LWR@siunitx@textprime} ,
    angle-symbol-second = {\LWR@siunitx@textdblprime} ,
}
}
```

Load late paches for lltjp-siunitx:

```
\AtBeginDocument{
\ifdef{\ltj@allalchar}
    {\LWR@origRequirePackage{lwarp-lltjp-siunitx}}
    {}
}
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-siunitx}

\CustomizeMathJax{\let\unit\si}
\CustomizeMathJax{\let\qty\SI}
\CustomizeMathJax{\let\qtylist\SIlist}
\CustomizeMathJax{\let\qtyrange\SIrange}
\CustomizeMathJax{\let\numproduct\num}
\CustomizeMathJax{\let\qtyproduct\SI}
\CustomizeMathJax{\let\complexnum\num}
\CustomizeMathJax{\newcommand{\complexqty}[3][]{(\complexnum{#2})\si{#3}}}
```

Pass `range-phrase` to common-mathjax-siunitx:

```
\ExplSyntaxOn
\AtBeginDocument{
\edef\LWR@siunitx@rangephrase{\l_siunitx_range_phrase_tl}
\expandafter\CustomizeMathJax\expandafter{%
    \expandafter\def\expandafter\LWRsiunitxrangephrase%
    \expandafter{\LWR@siunitx@rangephrase}%
}
}
\ExplSyntaxOff
```

File 453 **lwarp-siunitx-v2.sty**

§ 562 Package

siunitx-v2

(*Emulates or patches code by* Joseph Wright.)

siunitx-v2 (*Pkg*) siunitx-v2 is patched for use by lwarp, and is emulated for MathJax.

siunitx is well supported by lwarp.

Limitations Some general limitations:

fractions Due to *pdftotext* limitations, fraction output is replaced by symbol output for per-mode and quotient-mode.

\cancel is not currently supported for siunitx v3.

Negative values are not automatically colored.

⚠ tabular Tabular S and s columns are rendered as simple c columns, although key settings will be set. If using scientific notation, table-format, table-align-uncertainty, drop-exponent, etc.. use \tablenum for each cell. This is especially required for

⚠ drop-exponent drop-exponent, without which the value will be shown incorrectly.

⚠ table-auto-round table-auto-round is ignored.

Math rendering Math may be rendered in several ways in the same document:

For math mode with SVG display: The original siunitx code is used while generating the SVG image.

For HTML text mode: lwarp uses siunitx code patched for HTML, and simplified units.

For math expressions while using MathJax: A limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. The result usually looks fine, and otherwise is enough to get the meaning across.

Custom units siunitx allows customized units:

\DeclareSIUnit {⟨*name*⟩} {⟨*definition*⟩}

\DeclareSIUnit declares a version of the unit for the print version. This is also used when the unit is printed in SVG math or a lateximage. It is also used for HTML if an HTML-specific version is not defined with \HTMLDeclareSIUnit.

```
\DeclareSIUnit\myunit{\ensuremath{\text{m}_y}}
```

\HTMLDeclareSIUnit {⟨*name*⟩} {⟨*definition*⟩}

⚠ v3 only! Use this after the print unit has been defined. For siunitx v3, \HTMLDeclareSIUnit

declares a simplified version of the unit for HTML, for example if the print-mode unit uses TEX boxes or \ensuremath:

```
\HTMLDeclareSIUnit\myunit{\text{m}\textsubscript{\textit{y}}}
```

It is also possible to provide a custom unit for MathJax:

```
\CustomizeMathJax{\newcommand{\myunit}{\text{m}_y}}
```

Predefined units Most units work as-is with HTML. For the following units, lwarp has already set \HTMLDeclareSIUnit: \celsius, \arcminute, \arcsecond, \elementarycharge, \clight, \bohr, \electronmass, \hartree, \planckbar.

⚠ MathJax

Document modifications required for MathJax

⚠ \sisetup

- Place \sisetup in the preamble before \begin{document}. Changes made later may be ignored, especially with MathJax. The MathJax emulation also ignores most macro options.

⚠ complex numbers

- Complex numbers are displayed as entered, ignoring output-complex-root.

custom units

- Custom units may be added with \CustomizeMathJax. For example, from lwarp-common-mathjax-siunitx:

```
\CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
\CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
```

⚠ unit spacing

- Units work better using ~ between units instead of using periods.

⚠ \square, \cubic

- To square or cube compound units, enclose the following compound units in braces:

```
\cubic{\centi\meter}
```

  Single units do not require braces.

- For \numlist, the argument is printed as text as-is, so use space between semicolons for improved readability.

⚠ Missing $ inserted

- If using parse-numbers = false, also use \num or \qty. siunitx=siunitx>Missing $ inserted.

Also see MathJax option, section 8.7.5.

for HTML output:

```
\RequirePackage{xcolor}% for \convertcolorspec

\LWR@ProvidesPackagePass{siunitx}[=v2]% 2021-04-17


\AtBeginDocument{% in case textcomp was not loaded
    \DeclareSIUnit\bohr{\textit{a}\textsubscript{0}}
    \DeclareSIUnit\clight{\textit{c}\textsubscript{0}}
    \DeclareSIUnit\elementarycharge{\textit{e}}
    \DeclareSIUnit\electronmass{\textit{m}\textsubscript{e}}
    \DeclareSIUnit\hartree{\textit{E}\textsubscript{h}}
    \DeclareSIUnit\planckbar{\LWR@siunitx@textplanckbar}
}% AtBeginDocument
```

Support the S and s column types:

```
\AtBeginDocument{
\HTMLnewcolumntype{S}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
\HTMLnewcolumntype{s}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
}
```

\@ensuredmath is not supported inside an \hbox, so it must temporarily be restored to its original. Similar for \mbox. SVG math is created explicitly when necessary, using \LWR@subsingledollar.

```

\ExplSyntaxOn
%
```

Modified to use the print version of \@ensuredmath to avoid having a lateximage each time.

```
\AtBeginDocument{
\cs_set_protected:Npn \__siunitx_print_text:
  {
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath%          lwarp
    \tl_replace_all:Nnn \l__siunitx_print_arg_tl { - }
      { \textminus }
    \__siunitx_print_text_aux:
    \tl_replace_all:Nnn \l__siunitx_print_arg_tl { \mp }
      { \ensuremath { \mp } }
    \tl_remove_all:Nn \l__siunitx_print_arg_tl { \mathord }
    \cs_set_eq:NN \PrintSubscript   \__siunitx_print_text_sub:n
    \cs_set_eq:NN \PrintSuperscript \__siunitx_print_text_super:n
    \__siunitx_print_text_aux:NnN
      _ { math_subscript }   \__siunitx_print_text_sub:n
      _ { active }           \__siunitx_print_text_sub:n
      ^ { math_superscript } \__siunitx_print_text_super:n
      ^ { active }           \__siunitx_print_text_super:n
      \q_recursion_tail ?  ?
      \q_recursion_stop
    \l__siunitx_print_arg_tl
  }
}
```

Modified to set set HTML \textcolor if not black:

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_print_aux:
  {
    \text
      {
        \__siunitx_ensure_ltr:n
          {
            \color@begingroup
%
            \__siunitx_print_color:
            \__siunitx_font_shape:
            \__siunitx_font_weight:
            \use:c
              {
                __siunitx_ \l__siunitx_print_type_tl _
                text \l__siunitx_font_family_tl :
              }
```

```
%             \bool_if:NTF \l__siunitx_font_math_mode_bool
%               {
%                     \__siunitx_print_math:
%               }
                {
                     \LWR@findcurrenttextcolor% lwarp
                     \ifdefstring{\LWR@tempcolor}{000000}% lwarp
                         {\__siunitx_print_text:}% lwarp
                         {% lwarp
                             \LWR@textcurrentcolor{% lwarp
                                 \__siunitx_print_text:
                             }% lwarp
                         }% lwarp
                }
            \color@endgroup
%
          }
      }
  }
\LWR@formatted{__siunitx_print_aux:}

\cs_new_protected:Npn \LWR@HTML@__siunitx_set_math_fam:n #1 {
  \group_begin:
%     \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
%     \LetLtxMacro\mbox\LWR@print@mbox% lwarp
%     \hbox_set:Nn \l__siunitx_tmp_box
%       {
        \ensuremath
          {
            \use:c { math #1 }
              {
                \int_const:cn { c__siunitx_math #1 _int } { \fam }
              }
          }
%       }
  \group_end:
}
\LWR@formatted{__siunitx_set_math_fam:n}

\cs_new_protected:Npn \LWR@HTML@__siunitx_combined_output:n #1 {
%
    \group_begin:% lwarp
  \bool_if:NTF \l__siunitx_number_parse_bool
    {
      \tl_clear:N \l__siunitx_number_out_tl
      \bool_set_false:N \l__siunitx_number_compound_bool
      \__siunitx_number_output_parse:n {#1}
    }
    {
```

For parse-numbers=false:

```
        \__siunitx_unit_output_pre_print:
        \begingroup%                              lwarp
            \boolfalse{mathjax}%                  lwarp
%       \__siunitx_print:nn { number } { \ensuremath {#1} }
            \LWR@subsingledollar%       lwarp
                {%  alt text
                    \textbackslash( % space
                    \LWR@HTMLsanitizedetokenized{%
```

```
                        \detokenize{#1}%
                    } \textbackslash)%          lwarp
                }
                {siunitx}%  addl hashing
                {%
                    \__siunitx_print:nn { number } {%
                        \LWR@origensuredmath{#1}%
                    }%
                }%                              lwarp
        \endgroup%                              lwarp
        \__siunitx_unit_output_print:
    }
   \group_end:% lwarp
%
}
\LWR@formatted{__siunitx_combined_output:n}
```

For parse-numbers=false:

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_range_numbers_aux:n #1
  {
    \bool_if:NTF \l__siunitx_number_parse_bool
      {
        \tl_clear:N \l__siunitx_number_out_tl
        \tl_clear:N \l__siunitx_number_out_saved_tl
        \bool_set_false:N \l__siunitx_number_compound_bool
        \__siunitx_number_output_parse:n {#1}
        \bool_if:NT \l__siunitx_number_compound_bool
          { \msg_error:nnx { siunitx } { multi-part-range } {#1} }
      }
      {
        \__siunitx_unit_output_pre_print:
        \begingroup%    lwarp
            \boolfalse{mathjax}%                        lwarp
%           \__siunitx_print:nn { number } {#1}
                \LWR@subsingledollar%                   lwarp
                    {%  alt text
                        \textbackslash( % space
                        \LWR@HTMLsanitizedetokenized{%
                            \detokenize{#1}%
                        } \textbackslash)%              lwarp
                    }%
                    {siunitx}%  addl hashing
                    {%
                        \__siunitx_print:nn { number } {%
                            \LWR@origensuredmath{#1}%
                        } %                             lwarp
                    }%                                  lwarp
        \endgroup%                                      lwarp
        \__siunitx_unit_output_print:
      }
  }
\LWR@formatted{__siunitx_range_numbers_aux:n}
```

For parse-numbers=false:

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_angle_print_direct_aux:nn #1#2 {
  \tl_if_empty:nF {#1}
    {
      \tl_set:Nn \l__siunitx_unit_tl {#2}
```

```
        \begingroup%                                        lwarp
            \boolfalse{mathjax}%                            lwarp
%           \__siunitx_print:nn { number } {#1}
                \LWR@subsingledollar{%                      lwarp
                    \textbackslash( % space
                    \LWR@HTMLsanitizedetokenized{%
                        \detokenize{#1}%
                    } \textbackslash)%                      lwarp
                }%
                {siunitx}%
                {%
                    \__siunitx_print:nn { number } {
                        \LWR@origensuredmath{#1}%
                    }%                                      lwarp
                }%                                          lwarp
        \endgroup%                                          lwarp
      \__siunitx_unit_output_print:
    }
}
\LWR@formatted{__siunitx_angle_print_direct_aux:nn}
%
```

For quotients, the fraction code is replaced by the symbol code:

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_number_output_quotient_fraction: {
  \bool_set_true:N \l__siunitx_number_compound_bool
  \__siunitx_number_output_quotient_aux_i:
  \tl_set_eq:NN \l__siunitx_number_out_tl
    \l__siunitx_number_numerator_tl
  \tl_put_right:NV \l__siunitx_number_out_tl \l__siunitx_output_quotient_tl
  \tl_put_right:NV \l__siunitx_number_out_tl
    \l__siunitx_number_denominator_tl
  \__siunitx_number_output_single_aux:
}
\LWR@formatted{__siunitx_number_output_quotient_fraction:}
```

For units, the fraction code is replaced by the symbol code:

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_unit_format_fraction_fraction: {
  \__siunitx_unit_format_fraction_symbol_aux:
  \int_compare:nNnT { \l__siunitx_unit_denominator_int } > { 1 }
    {
      \bool_if:NT \l__siunitx_unit_denominator_bracket_bool
        {
       \tl_put_left:NV \l__siunitx_unit_denominator_tl \l__siunitx_bracket_open_tl
       \tl_put_right:NV \l__siunitx_unit_denominator_tl \l__siunitx_bracket_close_tl
        }
    }
  \tl_set_eq:NN \l__siunitx_unit_tl \l__siunitx_unit_numerator_tl
  \tl_put_right:NV \l__siunitx_unit_tl \l__siunitx_per_symbol_tl
  \tl_put_right:NV \l__siunitx_unit_tl \l__siunitx_unit_denominator_tl
}
\LWR@formatted{__siunitx_unit_format_fraction_fraction:}


\cs_new_protected:Npn \LWR@HTML@__siunitx_angle_print_astronomy_aux: {
  \prop_get:NnNT \l__siunitx_number_out_prop { mantissa-integer }
    \l__siunitx_tmpa_tl
    { \__siunitx_print:nV { number } \l__siunitx_tmpa_tl }
 \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}% lwarp
```

```
{% lateximage
 \hbox_set:Nn \l__siunitx_angle_marker_box
   {
     \__siunitx_print:nn { number } { { \l__siunitx_output_decimal_tl } }
   }
 \hbox_set:Nn \l__siunitx_angle_unit_box
   {
     \__siunitx_print:nV { unit } \l__siunitx_unit_tl
     \skip_horizontal:n { -\scriptspace }
   }
 \__siunitx_angle_print_astronomy_aux:n { marker }
 \__siunitx_angle_print_astronomy_aux:n { unit }
 \hbox_set:Nn \l__siunitx_angle_marker_box
   {
     \box_use:N \l__siunitx_angle_marker_box
     \box_use:N \l__siunitx_angle_unit_box
   }
 \dim_compare:nNnTF
   { \l__siunitx_angle_marker_dim } > { \l__siunitx_angle_unit_dim }
   { \__siunitx_angle_print_astronomy_marker: }
   { \__siunitx_angle_print_astronomy_unit: }
}% lateximage
{% not a lateximage
     \__siunitx_print:nn { number } { { \l__siunitx_output_decimal_tl } }
     \__siunitx_print:nV { unit } \l__siunitx_unit_tl
}% not a lateximage
 \prop_get:NnNT \l__siunitx_number_out_prop { mantissa-decimal }
   \l__siunitx_tmpa_tl
   { \__siunitx_print:nV { number } \l__siunitx_tmpa_tl }
}
\LWR@formatted{__siunitx_angle_print_astronomy_aux:}


\cs_new_protected:Npn \LWR@HTML@__siunitx_textsuperscript:n #1 {\textsuperscript{#1}}
\LWR@formatted{__siunitx_textsuperscript:n}

\cs_new_eq:NN \LWR@HTML@__siunitx_print_text_super:n \textsuperscript
\LWR@formatted{__siunitx_print_text_super:n}

\cs_new_eq:NN \LWR@HTML@__siunitx_print_text_sub:n \textsubscript
\LWR@formatted{__siunitx_print_text_sub:n}
```

`\LWR@origenduresmath` is added here in case the user asks for `\mathrm`, etc. for `output-exponent-marker`.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_number_format_final_exponent: {
 \prop_get:NnN \l__siunitx_number_out_prop { exponent }
   \l__siunitx_tmpa_tl
 \tl_if_empty:NTF \l__siunitx_output_exponent_tl
   {
     \tl_set:Nx \l__siunitx_tmpa_tl
      { ^ { \exp_not:V \l__siunitx_tmpa_tl } }
     \tl_put_left:NV \l__siunitx_tmpa_tl \l__siunitx_exponent_base_tl
   }
   {
     \tl_set:Nx  \l__siunitx_tmpa_tl
       {
         \LWR@origensuredmath{%    lwarp
           \exp_not:V \l__siunitx_output_exponent_tl
         }%                        lwarp
```

```
          \exp_not:N \mathord
          \exp_not:V  \l__siunitx_tmpa_tl
        }
    }
  \prop_put:NnV \l__siunitx_number_out_prop { exponent-result }
    \l__siunitx_tmpa_tl
}
\LWR@formatted{__siunitx_number_format_final_exponent:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_number_format_final_combined: {
  \__siunitx_number_format_brackets:n { mantissa }
  \prop_get:NnN \l__siunitx_number_out_prop { mantissa-result }
    \l__siunitx_tmpa_tl
  \tl_if_empty:NT \l__siunitx_output_exponent_tl
    {
      \tl_put_right:Nx \l__siunitx_tmpa_tl
        {
          \exp_not:N \LWR@origensuredmath%       lwarp
            {
              \bool_if:NTF \l__siunitx_tight_bool
                { {  \exp_not:V \l__siunitx_exponent_product_tl } }
                { { } \exp_not:V \l__siunitx_exponent_product_tl { } }
            }
        }
    }
  \prop_get:NnN \l__siunitx_number_out_prop { exponent-result }
    \l__siunitx_tmpb_tl
  \tl_put_right:NV \l__siunitx_tmpa_tl \l__siunitx_tmpb_tl
  \prop_put:NnV \l__siunitx_number_out_prop { result }
    \l__siunitx_tmpa_tl
  \prop_put:Nnn \l__siunitx_number_out_prop
    { result-bracket-exponent } { true }
}
\LWR@formatted{__siunitx_number_format_final_combined:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_number_output_parts_aux: {
  \bool_if:NTF \l__siunitx_multi_repeat_bool
    {
      \prop_if_in:NnT \l__siunitx_number_out_prop { mantissa-result }
        {
          \__siunitx_number_output_parts_aux:n { mantissa }
          \__siunitx_number_output_parts_aux:n { complex }
        }
      \prop_get:NnNT \l__siunitx_number_out_prop { exponent-result }
        \l__siunitx_tmpa_tl
        {
          \prop_if_in:NnT \l__siunitx_number_out_prop { mantissa-result }
            {
              \tl_put_left:Nx \l__siunitx_tmpa_tl
                {
                  \exp_not:N \LWR@origensuredmath
                    {
                      \bool_if:NTF \l__siunitx_tight_bool
                        { {  \exp_not:V \l__siunitx_exponent_product_tl } }
```

```
                    { { } \exp_not:V \l__siunitx_exponent_product_tl { } }
                  }
              }
            \prop_put:NnV \l__siunitx_number_out_prop { exponent }
              \l__siunitx_tmpa_tl
          }
        \__siunitx_number_output_parts_print:n { exponent }
      }
  }
  { \__siunitx_number_output_single: }
}
\LWR@formatted{__siunitx_number_output_parts_aux:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_unit_output_print: {
  \int_compare:nNnF { \l__siunitx_unit_prefix_int } = { 0 }
    {
      \tl_set:Nx \l__siunitx_tmpa_tl
        {
          \bool_if:NTF \l__siunitx_tight_bool
            {
              \exp_not:N \LWR@origensuredmath%      lwarp
                { { \exp_not:V \l__siunitx_exponent_product_tl } }
            }
            {
              \exp_not:N \LWR@origensuredmath%      lwarp
                { { } \exp_not:V \l__siunitx_exponent_product_tl { } }
            }
          \int_use:N \l__siunitx_unit_prefix_base_int
          ^ { \int_use:N \l__siunitx_unit_prefix_int }
        }
      \__siunitx_print:nV { number } \l__siunitx_tmpa_tl
    }
  \tl_if_empty:NF \l__siunitx_unit_tl
    {
      \__siunitx_unit_output_number_sep:
      \__siunitx_print:nV { unit } \l__siunitx_unit_tl
    }
}
\LWR@formatted{__siunitx_unit_output_print:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_range_exponent:
  {
    \bool_if:NT \l__siunitx_process_fixed_bool
      {
        \tl_set_eq:NN \l__siunitx_tmpa_tl \l__siunitx_exponent_product_tl
        \bool_if:NT \l__siunitx_tight_bool
          {
            \tl_set:Nx \l__siunitx_tmpa_tl
              { \exp_not:N \mathord \exp_not:o \l__siunitx_tmpa_tl }
          }
        \tl_set:Nx \l__siunitx_tmpa_tl
          {
            \exp_not:N \LWR@origensuredmath {%      lwarp
                { } \exp_not:o \l__siunitx_tmpa_tl  { }
```

```
            }
            10 \exp_not:N \PrintSuperscript
              { \int_use:N \l__siunitx_process_fixed_int }
          }
        \__siunitx_print:nV { number } \l__siunitx_tmpa_tl
      }
  }
\LWR@formatted{__siunitx_range_exponent:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_table_print_S_reserved_exponent_product:
  {
    \tl_set_eq:NN \l__siunitx_tmpb_tl \l__siunitx_exponent_product_tl
    \bool_if:NT \l__siunitx_tight_bool
      {
        \tl_set:Nx \l__siunitx_tmpb_tl
          { \exp_not:N \mathord \exp_not:o \l__siunitx_tmpb_tl }
      }
    \tl_set:Nx \l__siunitx_tmpa_tl
      {
     \exp_not:N \LWR@origensuredmath { { } \exp_not:o \l__siunitx_tmpb_tl { } }
        \exp_not:o \l__siunitx_tmpa_tl
      }
  }
\LWR@formatted{__siunitx_table_print_S_reserved_exponent_product:}
```

\LWR@origensuredmath is added here to avoid using an image for the output product.

```
\cs_new_protected:Npn \LWR@HTML@__siunitx_number_output_product_aux: {
  \bool_set_true:N \l__siunitx_number_compound_bool
  \__siunitx_number_preprocess:V \l__siunitx_number_arg_tl
  \bool_if:NF \l__siunitx_error_bool
    {
      \tl_if_empty:NTF \l__siunitx_number_multi_tl
        { \__siunitx_number_output_parse_aux: }
        { \__siunitx_number_output_quotient: }
      \tl_if_empty:NF \l__siunitx_number_next_tl
        {
          \bool_if:NTF \l__siunitx_tight_bool
            {
              \__siunitx_print:nn { number }
                { \LWR@origensuredmath { \l__siunitx_output_product_tl } }
            }
            {
              \__siunitx_print:nn  { number }
            { \LWR@origensuredmath { { } \l__siunitx_output_product_tl { } } }
            }
          \__siunitx_number_output_parse:V \l__siunitx_number_next_tl
        }
    }
}
\LWR@formatted{__siunitx_number_output_product_aux:}
```

Used to detect the math font.

```
\cs_set_protected:Npn \__siunitx_set_math_fam:n #1 {
```

```
  \group_begin:
    \hbox_set:Nn \l__siunitx_tmp_box
      {
        \LWR@origensuredmath%       lwarp
          {
            \use:c { math #1 }
              {
                \int_const:cn { c__siunitx_math #1 _int } { \fam }
              }
          }
      }
  \group_end:
}
```

Force \text:

```
\cs_set_protected:Npn \__siunitx_range_numbers:nn #1#2
  {
    \__siunitx_range_numbers_aux:n {#1}
    \text{\l__siunitx_range_phrase_tl}%     lwarp
    \__siunitx_range_numbers_aux:n {#2}
  }
```

Force \text:

```
\cs_set_protected:Npn \__siunitx_range_unit:nnnn #1#2#3#4 {
  \__siunitx_unit_parse_options:nn {#1} {#2}
  \bool_if:NTF \l__siunitx_range_repeat_bool
    {
      \__siunitx_unit_in:nn {#1} {#2}
      \__siunitx_range_numbers_aux:n {#3}
      \text{\l__siunitx_range_phrase_tl}%       lwarp
      \__siunitx_range_numbers_aux:n {#4}
    }
    {
      \bool_if:NT \l__siunitx_process_fixed_bool
        { \bool_set_true:N \l__siunitx_process_drop_exponent_bool }
      \bool_if:NT \l__siunitx_range_brackets_bool
        { \__siunitx_print:nV { number } \l__siunitx_bracket_open_tl }
      \__siunitx_range_numbers:nn {#3} {#4}
      \bool_if:NT \l__siunitx_range_brackets_bool
        { \__siunitx_print:nV { number } \l__siunitx_bracket_close_tl }
      \__siunitx_range_exponent:
      \__siunitx_unit_output_number_sep:
      \__siunitx_unit_output:nn {#1} {#2}
    }
}
```

```
\ExplSyntaxOff
```

```
\AtBeginDocument{
\sisetup{
    detect-mode=true,
    per-mode=symbol,    % fraction is not seen by pdftotext
    text-celsius = {\LWR@siunitx@textcelsius},
    text-degree = {\LWR@siunitx@textdegree},
    text-arcminute = {\LWR@siunitx@textprime} ,
    text-arcsecond = {\LWR@siunitx@textdblprime} ,
}
```

```
}
\LWR@origRequirePackage{lwarp-common-mathjax-siunitx}
```

Passing `range-phrase` to common-mathjax-siunitx does not seem to work with v2 using translator as it does with v3 using translations. The `range-phrase` therefore is set to an en-dash.

```
\AtBeginDocument{
\CustomizeMathJax{\def\LWRsiunitxrangephrase{\unicode{x2013}}}
}
```

File 454 **lwarp-common-mathjax-siunitx.sty**

§ 563 Package

common-mathjax-siunitx

(*Emulates or patches code by* Joseph Wright.)

common-mathjax-siunitx (*Pkg*) common-mathjax-siunitx adds MathJax for siunitx and siunitx-v2.

for HTML output: MathJax

For MathJax.

The following runs much faster as separate `\CusomizeMathJax` calls instead of one single call.

```
\begin{warpMathJax}
\LWR@infoprocessingmathjax{siunitx}

\CustomizeMathJax{\newcommand{\tothe}[1]{^{#1}}}
\CustomizeMathJax{\newcommand{\raiseto}[2]{{#2}^{#1}}}
```

Used as an end marker when parsing values:

```
\CustomizeMathJax{\newcommand{\LWRsiunitxEND}{}}
```

\ang [⟨*options*⟩] {⟨*value*⟩}

```
\CustomizeMathJax{\def\LWRsiunitxang#1;#2;#3;#4\LWRsiunitxEND{%
    \ifblank{#1}{}{\num{#1}\degree}%
    \ifblank{#2}{}{\num{#2}^{\unicode{x2032}}}% \prime
    \ifblank{#3}{}{\num{#3}^{\unicode{x2033}}}% \dblprime
}}
\CustomizeMathJax{\newcommand{\ang}[2][]{\LWRsiunitxang#2;;;\LWRsiunitxEND}}
```

\num [⟨*options*⟩] {⟨*value*⟩}

`\num` handles optional powers (`e`, `E`, `d`, `D`), multiples (`x`), plus and minus, and period or comma decimal output.

To split the string, `\def` is used with parameter delimiters. When each of the following macros is used, extra delimiters are padded to the end of the arguments of each macro when used, and the final argument of each collects any extra unused delimiters.

The number is split by dimensions (`x`), then by powers (`E`, `e`, `D`, `d`), then by plus / minus (`+-`, `\pm`), then by plus and minus (`+`, `-`), then into pieces before and after the decimal point or decimal comma.

Determine if the number is output with a decimal period or a decimal comma. The enclosing braces tell MathJax to not add extra space after the punctuation.

```
\ExplSyntaxOn
\AtBeginDocument{
\ifdefstring{\l__siunitx_output_decimal_tl}{{,}}
    {\CustomizeMathJax{\def\LWRsiunitxdecimal{,}}}
    {\CustomizeMathJax{\def\LWRsiunitxdecimal{.}}}
}
\ExplSyntaxOff
```

Any units which must be distributed across multiple dimensions:

```
\CustomizeMathJax{\def\LWRsiunitxdistribunit{}}
```

siunitx accepts either commas or periods as decimal points. `\LWRsiunitxprintdecimal` splits its input by periods then commas, parsing out before and after sections to print on either side of the decimal point.

`\LWRsiunitxENDTWO` is used only by `\LWRsiunitxprintdecimalsubtwo`, to avoid a parsing conflict with the more widely-used `\LWRsiunitxEND`.

The following splits by decimal commas:

```
\CustomizeMathJax{\newcommand{\LWRsiunitxENDTWO}{}}

\CustomizeMathJax{\def\LWRsiunitxprintdecimalsubtwo#1,#2,#3\LWRsiunitxENDTWO{%
```

If nothing is ahead of the decimal comma, add a leading zero:

```
    \ifblank{#1}{0}{\mathrm{#1}}%
```

If something is after the decimal comma, print the decimal and the fraction:

```
    \ifblank{#2}%
        {}%
        {%
            {\LWRsiunitxdecimal}%
            \mathrm{#2}%
        }%
}}
```

The following splits by decimal periods:

```
\CustomizeMathJax{\def\LWRsiunitxprintdecimalsub#1.#2.#3\LWRsiunitxEND{%
    \LWRsiunitxprintdecimalsubtwo#1,,\LWRsiunitxENDTWO%
    \ifblank{#2}%
        {}%
        {%
            {\LWRsiunitxdecimal}%
            \LWRsiunitxprintdecimalsubtwo#2,,\LWRsiunitxENDTWO%
        }%
}}

\CustomizeMathJax{\newcommand{\LWRsiunitxprintdecimal}[1]{%
    \LWRsiunitxprintdecimalsub#1...\LWRsiunitxEND%
}}
```

The following splits by +

```
\CustomizeMathJax{\def\LWRsiunitxnumplus#1+#2+#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {%
            \LWRsiunitxprintdecimal{#1}%
        }% no plus
        {%
```

```
            \ifblank{#1}%
                {\LWRsiunitxprintdecimal{#2}}% leading plus, ignore
                {% a+b
                    \LWRsiunitxprintdecimal{#1}%
                    \unicode{x02B}% plus sign
                    \LWRsiunitxprintdecimal{#2}%
                }%
        }%
    \LWRsiunitxdistribunit%
}}
```

The following splits by -

```
\CustomizeMathJax{\def\LWRsiunitxnumminus#1-#2-#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumplus#1+++\LWRsiunitxEND}%
        {%
            \ifblank{#1}{}{\LWRsiunitxprintdecimal{#1}}%
            \unicode{x02212}% mathematical minus sign
            \LWRsiunitxprintdecimal{#2}%
            \LWRsiunitxdistribunit%
        }%
}}
```

The following splits by \pm

```
\CustomizeMathJax{\def\LWRsiunitxnumpmmacro#1\pm#2\pm#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumminus#1---\LWRsiunitxEND}%
        {%
            \LWRsiunitxprintdecimal{#1}%
            \unicode{x0B1}% \pm
            \LWRsiunitxprintdecimal{#2}%
            \LWRsiunitxdistribunit%
        }%
}}
```

The following splits by +-

```
\CustomizeMathJax{\def\LWRsiunitxnumpm#1+-#2+-#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumpmmacro#1\pm\pm\pm\LWRsiunitxEND}%
        {%
            \LWRsiunitxprintdecimal{#1}%
            \unicode{x0B1}% \pm
            \LWRsiunitxprintdecimal{#2}%
            \LWRsiunitxdistribunit%
        }%
}}
```

Processes scientific notation. Special handling for a mantissa which is either empty or only a minus sign.

```
\CustomizeMathJax{\newcommand{\LWRsiunitxnumscientific}[2]{%
    \ifblank{#1}%
        {}%
        {%
            \ifstrequal{#1}{-}%
                {-}%
                {\LWRsiunitxprintdecimal{#1}\times}%
        }%
    10^{\LWRsiunitxprintdecimal{#2}}%
    \LWRsiunitxdistribunit%
}}
```

The following splits by D

```
\CustomizeMathJax{\def\LWRsiunitxnumD#1D#2D#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumpm#1+-+-\LWRsiunitxEND}%
        {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
}}
```

The following splits by d

```
\CustomizeMathJax{\def\LWRsiunitxnumd#1d#2d#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumD#1DDD\LWRsiunitxEND}%
        {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
}}
```

The following splits by E

```
\CustomizeMathJax{\def\LWRsiunitxnumE#1E#2E#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumd#1ddd\LWRsiunitxEND}%
        {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
}}
```

The following splits by e

```
\CustomizeMathJax{\def\LWRsiunitxnume#1e#2e#3\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnumE#1EEE\LWRsiunitxEND}%
        {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
}}
```

The following splits by x

```
\CustomizeMathJax{\def\LWRsiunitxnumx#1x#2x#3x#4\LWRsiunitxEND{%
    \ifblank{#2}%
        {\LWRsiunitxnume#1eee\LWRsiunitxEND}%
        {%
            \ifblank{#3}%
                {%
                    \LWRsiunitxnume#1eee\LWRsiunitxEND%
                    \times%
                    \LWRsiunitxnume#2eee\LWRsiunitxEND%
                }%
                {%
                    \LWRsiunitxnume#1eee\LWRsiunitxEND%
                    \times%
                    \LWRsiunitxnume#2eee\LWRsiunitxEND%
                    \times%
                    \LWRsiunitxnume#3eee\LWRsiunitxEND%
                }%
        }%
}}

\CustomizeMathJax{\newcommand{\num}[2][]{%
    \LWRsiunitxnumx#2xxxxx\LWRsiunitxEND%
}}
```

\si [⟨*options*⟩] {⟨*unit*⟩}

~ is converted to a thin space. Not able to convert period to thin space because the period might be in \raiseto, for example.

```
\CustomizeMathJax{\newcommand{\si}[2][]{%
```

```
    \mathrm{\gsubstitute{#2}{~}{\,}}%
}}
```

\SI [⟨*options*⟩] {⟨*value*⟩} [⟨*prefix*⟩] {⟨*unit*⟩}

\SI has a second optional arg, which is parsed using \ifnextchar.

```
\CustomizeMathJax{\def\LWRsiunitxSIopt#1[#2]#3{%
    \def\LWRsiunitxdistribunit{\,\si{#3}}%
    {#2}\num{#1}%
    \def\LWRsiunitxdistribunit{}%
}}

\CustomizeMathJax{\newcommand{\LWRsiunitxSI}[2]{%
    \def\LWRsiunitxdistribunit{\,\si{#2}}%
    \num{#1}%
    \def\LWRsiunitxdistribunit{}%
}}

\CustomizeMathJax{\newcommand{\SI}[2][]{%
    \ifnextchar[%
        {\LWRsiunitxSIopt{#2}}%
        {\LWRsiunitxSI{#2}}%
}}
```

\numlist [⟨*options*⟩] {⟨*list*⟩}

\numlist should only be used in text mode. If used in MathJax, it is merely printed as text, so add space around the semicolons.

```
\CustomizeMathJax{\newcommand{\numlist}[2][]{\text{#2}}}
```

\numrange [⟨*options*⟩] {⟨*value1*⟩} {⟨*value2*⟩}

\numrange should only be used in text mode. If used in MathJax math, an en-dash is used instead of the range-phrase.

```
\CustomizeMathJax{\newcommand{\numrange}[3][]{%
    \num{#2}\ \LWRsiunitxrangephrase\ \num{#3}%
}}
```

\SIlist [⟨*options*⟩] {⟨*list*⟩}

\SIlist and \SIrange should only be used in text mode. If used in MathJax, a simple emulation is provided.

```
\CustomizeMathJax{\newcommand{\SIlist}[3][]{\text{#2}\,\si{#3}}}
```

\SIrange [⟨*options*⟩] {⟨*value1*⟩} {⟨*value2*⟩} {⟨*unit*⟩}

```
\CustomizeMathJax{\newcommand{\SIrange}[4][]{%
    \num{#2}\,#4\ \LWRsiunitxrangephrase\ \num{#3}\,#4%
}}
```

\tablenum [⟨*options*⟩] {⟨*value*⟩}

```
\CustomizeMathJax{\newcommand{\tablenum}[2][]{\mathrm{#2}}}
```

```
\CustomizeMathJax{\newcommand{\ampere}{\mathrm{A}}}
\CustomizeMathJax{\newcommand{\candela}{\mathrm{cd}}}
\CustomizeMathJax{\newcommand{\kelvin}{\mathrm{K}}}
```

```
\CustomizeMathJax{\newcommand{\kilogram}{\mathrm{kg}}}
\CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\mole}{\mathrm{mol}}}
\CustomizeMathJax{\newcommand{\second}{\mathrm{s}}}
%
\CustomizeMathJax{\newcommand{\becquerel}{\mathrm{Bq}}}
\CustomizeMathJax{\newcommand{\degreeCelsius}{\unicode{x2103}}}
\CustomizeMathJax{\newcommand{\coulomb}{\mathrm{C}}}
\CustomizeMathJax{\newcommand{\farad}{\mathrm{F}}}
\CustomizeMathJax{\newcommand{\gray}{\mathrm{Gy}}}
\CustomizeMathJax{\newcommand{\hertz}{\mathrm{Hz}}}
\CustomizeMathJax{\newcommand{\henry}{\mathrm{H}}}
\CustomizeMathJax{\newcommand{\joule}{\mathrm{J}}}
\CustomizeMathJax{\newcommand{\katal}{\mathrm{kat}}}
\CustomizeMathJax{\newcommand{\lumen}{\mathrm{lm}}}
\CustomizeMathJax{\newcommand{\lux}{\mathrm{lx}}}
\CustomizeMathJax{\newcommand{\newton}{\mathrm{N}}}
\CustomizeMathJax{\newcommand{\ohm}{\mathrm{\Omega}}}
\CustomizeMathJax{\newcommand{\pascal}{\mathrm{Pa}}}
\CustomizeMathJax{\newcommand{\radian}{\mathrm{rad}}}
\CustomizeMathJax{\newcommand{\siemens}{\mathrm{S}}}
\CustomizeMathJax{\newcommand{\sievert}{\mathrm{Sv}}}
\CustomizeMathJax{\newcommand{\steradian}{\mathrm{sr}}}
\CustomizeMathJax{\newcommand{\tesla}{\mathrm{T}}}
\CustomizeMathJax{\newcommand{\volt}{\mathrm{V}}}
\CustomizeMathJax{\newcommand{\watt}{\mathrm{W}}}
\CustomizeMathJax{\newcommand{\weber}{\mathrm{Wb}}}
\CustomizeMathJax{\newcommand{\day}{\mathrm{d}}}
\CustomizeMathJax{\newcommand{\degree}{\mathrm{^\circ}}}
\CustomizeMathJax{\newcommand{\hectare}{\mathrm{ha}}}
\CustomizeMathJax{\newcommand{\hour}{\mathrm{h}}}
\CustomizeMathJax{\newcommand{\litre}{\mathrm{l}}}
\CustomizeMathJax{\newcommand{\liter}{\mathrm{L}}}
\CustomizeMathJax{\newcommand{\arcminute}{^\prime}}
\CustomizeMathJax{\newcommand{\minute}{\mathrm{min}}}
\CustomizeMathJax{\newcommand{\arcsecond}{^{\prime\prime}}}
\CustomizeMathJax{\newcommand{\tonne}{\mathrm{t}}}
\CustomizeMathJax{\newcommand{\astronomicalunit}{au}}
\CustomizeMathJax{\newcommand{\atomicmassunit}{u}}
\CustomizeMathJax{\newcommand{\bohr}{\mathit{a}_0}}
\CustomizeMathJax{\newcommand{\clight}{\mathit{c}_0}}
\CustomizeMathJax{\newcommand{\dalton}{\mathrm{D}_\mathrm{a}}}
\CustomizeMathJax{\newcommand{\electronmass}{\mathit{m}_{\mathrm{e}}}}
\CustomizeMathJax{\newcommand{\electronvolt}{\mathrm{eV}}}
\CustomizeMathJax{\newcommand{\elementarycharge}{\mathit{e}}}
\CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
\CustomizeMathJax{\newcommand{\planckbar}{\mathit{\unicode{x210F}}}}
\CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
\CustomizeMathJax{\let\LWRorigbar\bar}
\CustomizeMathJax{\newcommand{\bar}{\mathrm{bar}}}
\CustomizeMathJax{\newcommand{\barn}{\mathrm{b}}}
\CustomizeMathJax{\newcommand{\bel}{\mathrm{B}}}
\CustomizeMathJax{\newcommand{\decibel}{\mathrm{dB}}}
\CustomizeMathJax{\newcommand{\knot}{\mathrm{kn}}}
\CustomizeMathJax{\newcommand{\mmHg}{\mathrm{mmHg}}}
\CustomizeMathJax{\newcommand{\nauticalmile}{\mathrm{M}}}
\CustomizeMathJax{\newcommand{\neper}{\mathrm{Np}}}
%
\CustomizeMathJax{\newcommand{\yocto}{\mathrm{y}}}
\CustomizeMathJax{\newcommand{\zepto}{\mathrm{z}}}
```

```
\CustomizeMathJax{\newcommand{\atto}{\mathrm{a}}}
\CustomizeMathJax{\newcommand{\femto}{\mathrm{f}}}
\CustomizeMathJax{\newcommand{\pico}{\mathrm{p}}}
\CustomizeMathJax{\newcommand{\nano}{\mathrm{n}}}
\CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
\CustomizeMathJax{\newcommand{\milli}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\centi}{\mathrm{c}}}
\CustomizeMathJax{\newcommand{\deci}{\mathrm{d}}}
\CustomizeMathJax{\newcommand{\deca}{\mathrm{da}}}
\CustomizeMathJax{\newcommand{\hecto}{\mathrm{h}}}
\CustomizeMathJax{\newcommand{\kilo}{\mathrm{k}}}
\CustomizeMathJax{\newcommand{\mega}{\mathrm{M}}}
\CustomizeMathJax{\newcommand{\giga}{\mathrm{G}}}
\CustomizeMathJax{\newcommand{\tera}{\mathrm{T}}}
\CustomizeMathJax{\newcommand{\peta}{\mathrm{P}}}
\CustomizeMathJax{\newcommand{\exa}{\mathrm{E}}}
\CustomizeMathJax{\newcommand{\zetta}{\mathrm{Z}}}
\CustomizeMathJax{\newcommand{\yotta}{\mathrm{Y}}}
%
\CustomizeMathJax{\newcommand{\percent}{\mathrm{\%}}}
%
\CustomizeMathJax{\newcommand{\meter}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
%
\CustomizeMathJax{\newcommand{\gram}{\mathrm{g}}}
\CustomizeMathJax{\newcommand{\kg}{\kilo\gram}}
\CustomizeMathJax{\newcommand{\of}[1]{_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\squared}{^2}}
\CustomizeMathJax{\newcommand{\square}[1]{\mathrm{#1}^2}}
\CustomizeMathJax{\newcommand{\cubed}{^3}}
\CustomizeMathJax{\newcommand{\cubic}[1]{\mathrm{#1}^3}}


\CustomizeMathJax{\newcommand{\per}{\,\mathrm{/}}}


\CustomizeMathJax{\newcommand{\celsius}{\unicode{x2103}}}
%
\CustomizeMathJax{\newcommand{\fg}{\femto\gram}}
\CustomizeMathJax{\newcommand{\pg}{\pico\gram}}
\CustomizeMathJax{\newcommand{\ng}{\nano\gram}}
\CustomizeMathJax{\newcommand{\ug}{\micro\gram}}
\CustomizeMathJax{\newcommand{\mg}{\milli\gram}}
\CustomizeMathJax{\newcommand{\g}{\gram}}
\CustomizeMathJax{\newcommand{\kg}{\kilo\gram}}
%
\CustomizeMathJax{\newcommand{\amu}{\mathrm{u}}}
%
\CustomizeMathJax{\newcommand{\pm}{\pico\metre}}
\CustomizeMathJax{\newcommand{\nm}{\nano\metre}}
\CustomizeMathJax{\newcommand{\um}{\micro\metre}}
\CustomizeMathJax{\newcommand{\mm}{\milli\metre}}
\CustomizeMathJax{\newcommand{\cm}{\centi\metre}}
\CustomizeMathJax{\newcommand{\dm}{\deci\metre}}
\CustomizeMathJax{\newcommand{\m}{\metre}}
\CustomizeMathJax{\newcommand{\km}{\kilo\metre}}
%
\CustomizeMathJax{\newcommand{\as}{\atto\second}}
\CustomizeMathJax{\newcommand{\fs}{\femto\second}}
\CustomizeMathJax{\newcommand{\ps}{\pico\second}}
\CustomizeMathJax{\newcommand{\ns}{\nano\second}}
```

```
\CustomizeMathJax{\newcommand{\us}{\micro\second}}
\CustomizeMathJax{\newcommand{\ms}{\milli\second}}
\CustomizeMathJax{\newcommand{\s}{\second}}
%
\CustomizeMathJax{\newcommand{\fmol}{\femto\mol}}
\CustomizeMathJax{\newcommand{\pmol}{\pico\mol}}
\CustomizeMathJax{\newcommand{\nmol}{\nano\mol}}
\CustomizeMathJax{\newcommand{\umol}{\micro\mol}}
\CustomizeMathJax{\newcommand{\mmol}{\milli\mol}}
\CustomizeMathJax{\newcommand{\mol}{\mol}}
\CustomizeMathJax{\newcommand{\kmol}{\kilo\mol}}
%
\CustomizeMathJax{\newcommand{\pA}{\pico\ampere}}
\CustomizeMathJax{\newcommand{\nA}{\nano\ampere}}
\CustomizeMathJax{\newcommand{\uA}{\micro\ampere}}
\CustomizeMathJax{\newcommand{\mA}{\milli\ampere}}
\CustomizeMathJax{\newcommand{\A}{\ampere}}
\CustomizeMathJax{\newcommand{\kA}{\kilo\ampere}}
%
\CustomizeMathJax{\newcommand{\ul}{\micro\litre}}
\CustomizeMathJax{\newcommand{\ml}{\milli\litre}}
\CustomizeMathJax{\newcommand{\l}{\litre}}
\CustomizeMathJax{\newcommand{\hl}{\hecto\litre}}
\CustomizeMathJax{\newcommand{\uL}{\micro\liter}}
\CustomizeMathJax{\newcommand{\mL}{\milli\liter}}
\CustomizeMathJax{\newcommand{\L}{\liter}}
\CustomizeMathJax{\newcommand{\hL}{\hecto\liter}}
%
\CustomizeMathJax{\newcommand{\mHz}{\milli\hertz}}
\CustomizeMathJax{\newcommand{\Hz}{\hertz}}
\CustomizeMathJax{\newcommand{\kHz}{\kilo\hertz}}
\CustomizeMathJax{\newcommand{\MHz}{\mega\hertz}}
\CustomizeMathJax{\newcommand{\GHz}{\giga\hertz}}
\CustomizeMathJax{\newcommand{\THz}{\tera\hertz}}
%
\CustomizeMathJax{\newcommand{\mN}{\milli\newton}}
\CustomizeMathJax{\newcommand{\N}{\newton}}
\CustomizeMathJax{\newcommand{\kN}{\kilo\newton}}
\CustomizeMathJax{\newcommand{\MN}{\mega\newton}}
%
\CustomizeMathJax{\newcommand{\Pa}{\pascal}}
\CustomizeMathJax{\newcommand{\kPa}{\kilo\pascal}}
\CustomizeMathJax{\newcommand{\MPa}{\mega\pascal}}
\CustomizeMathJax{\newcommand{\GPa}{\giga\pascal}}
%
\CustomizeMathJax{\newcommand{\mohm}{\milli\ohm}}
\CustomizeMathJax{\newcommand{\kohm}{\kilo\ohm}}
\CustomizeMathJax{\newcommand{\Mohm}{\mega\ohm}}
%
\CustomizeMathJax{\newcommand{\pV}{\pico\volt}}
\CustomizeMathJax{\newcommand{\nV}{\nano\volt}}
\CustomizeMathJax{\newcommand{\uV}{\micro\volt}}
\CustomizeMathJax{\newcommand{\mV}{\milli\volt}}
\CustomizeMathJax{\newcommand{\V}{\volt}}
\CustomizeMathJax{\newcommand{\kV}{\kilo\volt}}
%
\CustomizeMathJax{\newcommand{\W}{\watt}}
\CustomizeMathJax{\newcommand{\uW}{\micro\watt}}
\CustomizeMathJax{\newcommand{\mW}{\milli\watt}}
\CustomizeMathJax{\newcommand{\kW}{\kilo\watt}}
```

```
\CustomizeMathJax{\newcommand{\MW}{\mega\watt}}
\CustomizeMathJax{\newcommand{\GW}{\giga\watt}}
%
\CustomizeMathJax{\newcommand{\J}{\joule}}
\CustomizeMathJax{\newcommand{\uJ}{\micro\joule}}
\CustomizeMathJax{\newcommand{\mJ}{\milli\joule}}
\CustomizeMathJax{\newcommand{\kJ}{\kilo\joule}}
%
\CustomizeMathJax{\newcommand{\eV}{\electronvolt}}
\CustomizeMathJax{\newcommand{\meV}{\milli\electronvolt}}
\CustomizeMathJax{\newcommand{\keV}{\kilo\electronvolt}}
\CustomizeMathJax{\newcommand{\MeV}{\mega\electronvolt}}
\CustomizeMathJax{\newcommand{\GeV}{\giga\electronvolt}}
\CustomizeMathJax{\newcommand{\TeV}{\tera\electronvolt}}
%
\CustomizeMathJax{\newcommand{\kWh}{\kilo\watt\hour}}
%
\CustomizeMathJax{\newcommand{\F}{\farad}}
\CustomizeMathJax{\newcommand{\fF}{\femto\farad}}
\CustomizeMathJax{\newcommand{\pF}{\pico\farad}}
%
\CustomizeMathJax{\newcommand{\K}{\mathrm{K}}}
%
\CustomizeMathJax{\newcommand{\dB}{\mathrm{dB}}}
%
\CustomizeMathJax{\newcommand{\kibi}{\mathrm{Ki}}}
\CustomizeMathJax{\newcommand{\mebi}{\mathrm{Mi}}}
\CustomizeMathJax{\newcommand{\gibi}{\mathrm{Gi}}}
\CustomizeMathJax{\newcommand{\tebi}{\mathrm{Ti}}}
\CustomizeMathJax{\newcommand{\pebi}{\mathrm{Pi}}}
\CustomizeMathJax{\newcommand{\exbi}{\mathrm{Ei}}}
\CustomizeMathJax{\newcommand{\zebi}{\mathrm{Zi}}}
\CustomizeMathJax{\newcommand{\yobi}{\mathrm{Yi}}}
\end{warpMathJax}
```

File 455 **lwarp-skmath.sty**

§ 564 Package

skmath

(*Emulates or patches code by* Simon Sigurdhsson.)

skmath (*Pkg*) skmath is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{skmath}[2019/10/15]
```

Only defined if package option requested:

```
\begin{warpMathJax}
\ExplSyntaxOn
\bool_if:NT\g__skmath_define_common_sets_bool{
\CustomizeMathJax{\newcommand{\N}{\mathbb{N}}}
\CustomizeMathJax{\newcommand{\Z}{\mathbb{Z}}}
\CustomizeMathJax{\newcommand{\Q}{\mathbb{Q}}}
\CustomizeMathJax{\newcommand{\R}{\mathbb{R}}}
\CustomizeMathJax{\newcommand{\C}{\mathbb{C}}}
}
```

skmath is using l3keys, which does not seem to have an equivalent to `\@ifpackagewith`. To detect package options, comparisons with the following are made to see if various macros have been defined as follows:

```
\cs_gset_nopar:Npn\LWR__skmath_imaginary_unit:n#1{{#1}}
\cs_gset_nopar:Npn\LWR__skmath_natural_log_e:{{e}}
\cs_gset_nopar:Npn\LWR__skmath_integral_d:{{d}}
\cs_gset_nopar:Npn\LWR__skmath_total_derivative_d:{{d}}
```

If `notation=iso`, use upright, else italic:

```
\cs_if_eq:NNTF \__skmath_imaginary_unit:n \LWR__skmath_imaginary_unit:n
    {
        \CustomizeMathJax{\newcommand{ii}{\mathit{i}}}
        \CustomizeMathJax{\newcommand{jj}{\mathit{j}}}
    }
    {
        \CustomizeMathJax{\newcommand{ii}{\mathrm{i}}}
        \CustomizeMathJax{\newcommand{jj}{\mathrm{j}}}
    }
```

If `notation=iso`, use upright, else italic:

```
\cs_if_eq:NNTF \__skmath_natural_log_e: \LWR__skmath_natural_log_e:
    { \CustomizeMathJax{\newcommand{\ee}{\mathit{e}}} }
    { \CustomizeMathJax{\newcommand{\ee}{\mathrm{e}}} }
```

skmath uses `\DeclarePairedDelimiter` from mathtools for `\abs` and `\norm`, and lwarp uses this to automatically define MathJax definitions for each.

If `notation=english`, use slanted, else upright:

```
\cs_if_eq:NNTF \__skmath_integral_d: \LWR__skmath_integral_d:
    { \CustomizeMathJax{\newcommand{\d}{\mathit{d}}} }
    { \CustomizeMathJax{\newcommand{\d}{\mathrm{d}}} }
```

Used to parse comma and caret arguments for `\pd` and `\td`:

```
\CustomizeMathJax{\def\LWRskmathEND{}}
```

Parse the arguments with up to four commas. Argument 6 contains any leftover commas.

```
\CustomizeMathJax{\def\LWRskmathpdstarsub#1#2,#3,#4,#5,#6\LWRskmathEND{
    #1_{#2#3#4#5}%
}}

\CustomizeMathJax{\newcommand{\LWRskmathpdstar}[2]{%
    \LWRskmathpdstarsub{#1}#2,,,,\LWRskmathEND%
}}
```

Parse the arguments with up to two carets. Argument 3 contains any leftover carets. `\LWRskmathpdplus` is used to only place a plus sign starting after the first term. `\LWRskmathpdone` is used to only place a `1` digit if a second or later term does not have a power.

```
\CustomizeMathJax{\def\LWRskmathpdnumerator#1^#2^#3\LWRskmathEND{%
    \ifblank{#1}{}{
```

```
        \ifblank{#2}{\LWRskmathpdplus\LWRskmathpdone}{\LWRskmathpdplus#2}
    }
}}
```

Parse the arguments with up to two carets. Argument 3 contains any leftover carets.

```
\CustomizeMathJax{\def\LWRskmathpddenominator#1^#2^#3\LWRskmathEND{%
    \ifblank{#1}{}{%
        \ifblank{#2}%
            {\partial{#1}}%
            {\partial{#1}^{#2}}%
    }%
}}
```

Factored from `\LWRskmathpdnostarsub`, following:

The phrase `^{}` appears to be required while parsing the carets. `\LWRskmathpdplus` is used to only place a plus sign starting after the first term. `\LWRskmathpdone` is used to only place a `1` digit if a second or later term does not have a power.

This may not be recursion-safe. (Is there really such as a thing as nested differentials?)

```
\CustomizeMathJax{\newcommand{\LWRskmathdonumerator}[5]{%
    \partial^{%
        \def\LWRskmathpdplus{}%
        \LWRskmathpdnumerator#2^{}^{}\LWRskmathEND%
        \def\LWRskmathpdplus{+}%
        \def\LWRskmathpdone{1}%
        \LWRskmathpdnumerator#3^{}^{}\LWRskmathEND%
        \LWRskmathpdnumerator#4^{}^{}\LWRskmathEND%
        \LWRskmathpdnumerator#5^{}^{}\LWRskmathEND%
    }%
    {#1}%
}}

\CustomizeMathJax{\newcommand{\LWRskmathdodenominator}[4]{%
    \LWRskmathpddenominator#1^{}^{}^{}\LWRskmathEND%
    \ifblank{#2}{}{\,}%
    \LWRskmathpddenominator#2^{}^{}^{}\LWRskmathEND%
    \ifblank{#3}{}{\,}%
    \LWRskmathpddenominator#3^{}^{}^{}\LWRskmathEND%
    \ifblank{#4}{}{\,}%
    \LWRskmathpddenominator#4^{}^{}^{}\LWRskmathEND%
}}
```

Parse the arguments with up to four commas. Argument 6 contains any leftover commas.

```
\CustomizeMathJax{\def\LWRskmathpdnostarsub#1#2,#3,#4,#5,#6\LWRskmathEND{
    \ifblank{#3}{\def\LWRskmathpdone{}}{\def\LWRskmathpdone{1}}
    \frac%
        {\LWRskmathdonumerator{#1}{#2}{#3}{#4}{#5}}%
        {\LWRskmathdodenominator{#2}{#3}{#4}{#5}}%
}}

\CustomizeMathJax{\newcommand{\LWRskmathpdnostar}[2]{%
    \LWRskmathpdnostarsub{#1}#2,,,,,,\LWRskmathEND%
```

```
}}

\CustomizeMathJax{\newcommand{\pd}{\ifstar\LWRskmathpdstar\LWRskmathpdnostar}}
```

If `notation=english` or `legacy`, use slanted, else upright:

```
\cs_if_eq:NNTF \__skmath_total_derivative_d: \LWR__skmath_total_derivative_d:
    { \CustomizeMathJax{\newcommand{\LWRskmathtd}{\mathit{d}}} }
    { \CustomizeMathJax{\newcommand{\LWRskmathtd}{\mathrm{d}}} }

\CustomizeMathJax{\def\LWRskmathtdsub#1#2^#3\LWRskmathEND{%
    \frac
        {\LWRskmathtd^{#3}{#1}}
        {\LWRskmathtd{#2}^{#3}}
}}

\CustomizeMathJax{\newcommand{\td}[2]{%
    \LWRskmathtdsub{#1}#2^{}\LWRskmathEND%
}}

\CustomizeMathJax{\newcommand{\E}[1]{%
    \operatorname{E}\left[#1\right]%
}}

\CustomizeMathJax{\let\given\mid}

\CustomizeMathJax{\newcommand{\P}[1]{%
    \operatorname{P}%
    \left(#1\right)%
}}

\CustomizeMathJax{\newcommand{\var}[1]{%
  \operatorname{Var}\left(#1\right)%
}}

\CustomizeMathJax{\newcommand{\cov}[2]{%
  \operatorname{Cov}\left(#1,#2\right)%
}}
```

Common code for `\sin` etc:

```
\CustomizeMathJax{\newcommand{\LWRskmathtrigtwo}[2][]{%
    \ifblank{#1}{}{^{#1}}%
    \ifblank{#2}{}{\left(#2\right)}%
}}

\CustomizeMathJax{\newcommand{\LWRskmathtrig}[1]{%
    \operatorname{#1}%
    \LWRskmathtrigtwo%
}}

\CustomizeMathJax{\renewcommand{\sin}{\LWRskmathtrig{sin}}}
\CustomizeMathJax{\renewcommand{\arcsin}{\LWRskmathtrig{arcsin}}}

\CustomizeMathJax{\renewcommand{\cos}{\LWRskmathtrig{cos}}}
\CustomizeMathJax{\renewcommand{\arccos}{\LWRskmathtrig{arccos}}}

\CustomizeMathJax{\renewcommand{\tan}{\LWRskmathtrig{tan}}}
```

```
\CustomizeMathJax{\renewcommand{\arctan}{\LWRskmathtrig{arctan}}}

\CustomizeMathJax{\renewcommand{\cot}{\LWRskmathtrig{cot}}}

\CustomizeMathJax{\renewcommand{\sinh}{\LWRskmathtrig{sinh}}}
\CustomizeMathJax{\renewcommand{\cosh}{\LWRskmathtrig{cosh}}}
\CustomizeMathJax{\renewcommand{\tanh}{\LWRskmathtrig{tanh}}}
```

Common code for `\ln` and `\log`:

```
\CustomizeMathJax{\newcommand{\LWRskmathlogtwo}[2][]{%
    \ifblank{#1}{}{_{#1}}%
    \ifblank{#2}{}{\left(#2\right)}%
}}

\CustomizeMathJax{\newcommand{\LWRskmathlog}[1]{%
    \operatorname{#1}%
    \LWRskmathlogtwo%
}}

\CustomizeMathJax{\renewcommand{\ln}{\LWRskmathlog{ln}}}
\CustomizeMathJax{\renewcommand{\log}{\LWRskmathlog{log}}}

\CustomizeMathJax{\newcommand{\LWRskmathexpparens}[1]{%
    \operatorname{exp}%
    \ifblank{#1}{}{\left(#1\right)}%
}}
```

See the skmath source for the original of the following:

```
\CustomizeMathJax{\newcommand{\LWRskmathexpnostar}[1]{%
    \mathchoice
        {\ee^{#1}}
        {\LWRskmathexpparens{#1}}
        {\LWRskmathexpparens{#1}}
        {\LWRskmathexpparens{#1}}
}}

\CustomizeMathJax{\renewcommand{\exp}{\ifstar\LWRskmathexpparens\LWRskmathexpnostar}}
```

Common code for `\min` etc:

```
\CustomizeMathJax{\newcommand{\LWRskmathminstar}[2][]{%
    \operatorname{\LWRskmathminname}%
    \ifblank{#1}{}{%
        _{\mathchoice{\mathclap{#1}}{#1}{#1}{#1}}
    }%
    \ifblank{#2}{}{#2}%
}}

\CustomizeMathJax{\newcommand{\LWRskmathminnostar}[2][]{%
    \ifblank{#1}%
        {\operatorname{\LWRskmathminname}}%
        {%
            \underset%
                {\mathchoice{\mathclap{#1}}{#1}{#1}{#1}}%
                {\operatorname{\LWRskmathminname}}%
        }%
```

```
    \ifblank{#2}{}{\left\{#2\right\}}%
}}
```

\LWRskmathminname seems to be recursion-safe since it is used immediately.

```
\CustomizeMathJax{\newcommand{\LWRskmathmin}[1]{%
    \def\LWRskmathminname{#1}%
    \ifstar\LWRskmathminstar\LWRskmathminnostar%
}}

\CustomizeMathJax{\renewcommand{\min}{\LWRskmathmin{min}}}
\CustomizeMathJax{\renewcommand{\argmin}{\arg\LWRskmathmin{min}}}

\CustomizeMathJax{\renewcommand{\max}{\LWRskmathmin{max}}}
\CustomizeMathJax{\renewcommand{\argmax}{\arg\LWRskmathmin{max}}}
\CustomizeMathJax{\renewcommand{\sup}{\LWRskmathmin{sup}}}
\CustomizeMathJax{\renewcommand{\inf}{\LWRskmathmin{inf}}}

\CustomizeMathJax{\let\bar\overline}

\CustomizeMathJax{\let\vec\boldsymbol}
```

Remember the original definitions:

```
\CustomizeMathJax{\let\LWRskmathRe\Re}
\CustomizeMathJax{\let\LWRskmathIm\Im}
```

Redefine depending on notation=iso:

```
\bool_if:NTF\g__skmath_iso_complex_parts_bool{
    \CustomizeMathJax{\renewcommand{\Re}[1]{%
        \LWRskmathRe%
        \ifblank{#1}{}{\left(#1\right)}%
    }}
    \CustomizeMathJax{\renewcommand{\Im}[1]{%
        \LWRskmathIm%
        \ifblank{#1}{}{\left(#1\right)}%
    }}
}{
    \CustomizeMathJax{\renewcommand{\Re}[1]{%
        \operatorname{Re}%
        \ifblank{#1}{}{#1}%
    }}
    \CustomizeMathJax{\renewcommand{\Im}[1]{%
        \operatorname{Im}%
        \ifblank{#1}{}{#1}%
    }}
}

\ExplSyntaxOff
\end{warpMathJax}
```

File 456 **lwarp-slantsc.sty**

§ 565 Package slantsc

(*Emulates or patches code by* Harald Harders.)

slantsc (*Pkg*) slantsc is emulated for HTML, and used as-is for print output.

for HTML output:

```
\LWR@ProvidesPackagePass{slantsc}[2012/01/01]

\newcommand*{\LWR@HTML@noscshape}{}
\LWR@formatted{noscshape}

\FilenameNullify{%
    \LetLtxMacro\noscshape\@empty%
}
```

File 457 **lwarp-slashed.sty**

§ 566 Package slashed

(*Emulates or patches code by* David Carlisle.)

slashed (*Pkg*) slashed works as-s for HTML SVG math. For MathJax, emulation is provided.

for HTML output:

```
\LWR@ProvidesPackagePass{slashed}[1997/01/16]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\slashed}[1]{\cancel{#1}}}
\end{warpMathJax}
```

File 458 **lwarp-soul.sty**

§ 567 Package soul

(*Emulates or patches code by* Melchior FRANZ.)

soul (*Pkg*) soul is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{soul}[2003/11/17]
\RequirePackage{xcolor}% for \convertcolorspec
```

Storage for the colors to use:

```
\newcommand*{\LWR@soululcolor}{}

\newcommand*{\LWR@soulstcolor}{}

% \definecolor{LWR@soulhlcolordefault}{HTML}{F8E800}
% \newcommand*{\LWR@soulhlcolor}{LWR@soulhlcolordefault}
\newcommand*{\LWR@soulhlcolor}{}
```

\so {⟨*text*⟩}

Basic markup with CSS:

```
\newcommand{\so}[1]{%
\InlineClass(letter-spacing:.2ex){letterspacing}{#1}%
}
```

\caps {⟨*text*⟩}

```
\newcommand{\caps}[1]{%
    \InlineClass%
        (font-variant:small-caps;letter-spacing:.1ex)%
        {capsspacing}{#1}%
}
```

\LWR@soulcolor {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
\newcommand{\LWR@soulcolor}[5]{%
\ifcsempty{#2}%
{%
    \InlineClass(#5){#3}{#1}%
}%
{%
    \convertcolorspec{named}{\@nameuse{#2}}{HTML}\LWR@tempcolor%
    \LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}%
}%
}
```

```
\newcommand{\ul}[1]{%
\LWR@soulcolor{#1}{LWR@soululcolor}{uline}{text-decoration-color}%
    {text-decoration:underline; text-decoration-skip: auto;}%
}

\newcommand{\st}[1]{
\LWR@soulcolor{#1}{LWR@soulstcolor}{sout}{text-decoration-color}%
    {text-decoration:line-through}%
}

\newcommand{\hl}[1]{
\LWR@soulcolor{#1}{LWR@soulhlcolor}{highlight}{background-color}%
    {background:\LWR@origpound{}F8E800}
}
```

Nullified:

```
\newcommand*{\soulaccent}[1]{}
\newcommand*{\soulregister}[2]{}
\newcommand{\sloppyword}[1]{#1}
\newcommand*{\sodef}[5]{\DeclareRobustCommand*#1[1]{\so{##1}}}
\newcommand*{\resetso}{}
\newcommand*{\capsdef}[5]{}
\newcommand*{\capsreset}{}
\newcommand*{\capssave}[1]{}
\newcommand*{\capsselect}[1]{}
\newcommand*{\setul}[2]{}
\newcommand*{\resetul}{}
\newcommand*{\setuldepth}[1]{}
\newcommand*{\setuloverlap}[1]{}
\newcommand*{\<}{}
```

Set colors:

```
\newcommand*{\setulcolor}[1]{\renewcommand{\LWR@soululcolor}{#1}}
\newcommand*{\setstcolor}[1]{\renewcommand{\LWR@soulstcolor}{#1}}
```

```
\newcommand*{\sethlcolor}[1]{\renewcommand{\LWR@soulhlcolor}{#1}}
```

Long versions of the user-level macros:

```
\let\textso\so
\let\textul\ul
\let\texthl\hl
\let\textcaps\caps
```

File 459 **lwarp-soulpos.sty**

§ 568 Package soulpos

(*Emulates or patches code by* Javier Bezos.)

soulpos (*Pkg*) soulpos is emulated.

for HTML output:

```
\RequirePackage{soul}
\RequirePackage{soulutf8}
\LWR@ProvidesPackageDrop{soulpos}[2012/02/25]

\NewDocumentCommand{\ulposdef}{m o m}{}

\newdimen\ulwidth

\newcommand\ifulstarttype[1]{%
\expandafter\@secondoftwo%
}

\newcommand\ifulendtype[1]{%
\expandafter\@secondoftwo%
}

\newcommand{\ulstarttype}{0}
\newcommand{\ulendtype}{0}
\newcommand\ulpostolerance{0}%
```

File 460 **lwarp-soulutf8.sty**

§ 569 Package soulutf8

soulutf8 (*Pkg*) soulutf8 is emulated.

lwarp's HTML output naturally supports UTF-8 encoding.

for HTML output:

```
\LWR@ProvidesPackageDrop{soulutf8}[2016/05/16]
\RequirePackage{soul}
```

File 461 **lwarp-splitbib.sty**

§ 570 Package splitbib

(*Emulates or patches code by* Nicolas Markey.)

splitbib (*Pkg*) splitbib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{splitbib}[2005/12/22]

\def\NMSB@stylebox#1#2{%
\begin{BlockClass}[text-align:center ; border: 1px solid black]{splitbibbox}
    \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{#1#2}}
\end{BlockClass}
}

\def\NMSB@stylebar#1#2{%
\begin{BlockClass}[%
    text-align:center ;
    border-top: 1px solid black ;
    border-bottom: 1px solid black ;
]{splitbibbar}
    \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{#1#2}}
\end{BlockClass}
}

\def\NMSB@styledash#1#2{%
\begin{BlockClass}[%
    text-align:center ;
]{splitbibdash}
    \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{---~#1#2~---}}
\end{BlockClass}
}

\def\NMSB@stylenone#1#2{%
    \par
}

\def\NMSB@stylesimple#1#2{%
\par
 \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{#1#2}}
\par
}
```

File 462 **lwarp-splitidx.sty**

§ 571 Package splitidx

(*Emulates or patches code by* Markus Kohm.)

splitidx (*Pkg*) splitidx is patched for use by lwarp.

If the latexmk option is selected for lwarp, *latexmk* will compile the document but will *not* compile the indexes. **lwarpmk printindex** and **lwarpmk htmlindex** will still be required.

⚠ \thepage When using \AtWriteToIndex or \AtNextWriteToIndex, the user must not refer to \thepage during HTML output, as the concept of a page number is meaningless. Instead, do

```
\addtocounter{LWR@autoindex}{1}
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}
```

where the \index-like action occurs, and then refer to \arabic{LWR@autoindex} instead of \thepage where the reference should occur.

See section 695.17 in the lwarp-patch-memoir package for the \@@wrspindexhyp macro as an example.

for HTML output:

```
\LWR@ProvidesPackagePass{splitidx}[2016/02/18]

\catcode`\_=12%
\xpatchcmd{\newindex}
    {\jobname-#2.idx}
    {\jobname-#2_html.idx}
    {}
    {\LWR@patcherror{splitidx}{@newindex}}
\catcode`\_=8%
```

Patched to use lwarp's automatic indexing counter instead of \thepage:

```
\renewcommand*{\@wrsindex}[2][]{%
  \ifx\relax#1\relax
    \if@splitidx
      \@wrsindex[idx]{#2}%
    \else
      \def\@tempa{#2}%
      \if@verbindex\@onelevel@sanitize\@tempa\fi
      \@wrindex{\@tempa}%
    \fi
  \else
    \def\@tempa{#2}%
    \csname index@#1@hook\endcsname
%     \expandafter\ifx\csname @@wrsindex\endcsname\relax
    \addtocounter{LWR@autoindex}{1}%                    lwarp
%       \@@@wrsindex{#1}{{\@tempa}{\thepage}}%
    \@@@wrsindex{#1}{{\@tempa}{\arabic{LWR@autoindex}}}%
%     \else
%       \def\@tempb{\@@wrsindex{#1}}%
%       \expandafter\@tempb\@tempa||\\%
%     \fi
```

The label is assigned after the file write to avoid conflict with cleveref.

```
    \label{LWRindex-\arabic{LWR@autoindex}}%    lwarp
    \endgroup
    \@esphack
  \fi
}
```

lwarp defines sectioning commands with xparse, so the below patches are done as temporary redefinitions instead of being \let.

```
\xpatchcmd{\printsubindex}
    {\let\section\subsection}
    {\renewcommand*{\section}{\subsection}}
    {}
    {\LWR@patcherror{splitidx}{printsubindex-section}}

\xpatchcmd{\printsubindex}
    {\let\chapter\section}
    {\renewcommand*{\chapter}{\section}}
```

```
    {}
    {\LWR@patcherror{splitidx}{printsubindex-chapter}}

\xpatchcmd{\printsubindex}
    {\let\@makechapterhead\section}
    {\def\@makechapterhead{\section}}
    {}
    {\LWR@patcherror{splitidx}{printsubindex-chapter}}
```

File 463 **lwarp-srcltx.sty**

§ 572 Package srcltx

srcltx (*Pkg*) srcltx is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{srcltx}[2006/11/12]

\newif\ifSRCOK \SRCOKfalse
\newcommand*\srcIncludeHook[1]{}
\newcommand*\srcInputHook[1]{}
\newcommand*\MainFile{}
\def\MainFile{\jobname.tex}
\newcommand*\CurrentInput{}
\gdef\CurrentInput{\MainFile}
\newcommand\Input{}
\let\Input\input
```

File 464 **lwarp-srctex.sty**

§ 573 Package srctex

srctex (*Pkg*) srctex is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{srctex}[2006/11/12]
\LWR@origRequirePackage{lwarp-srcltx}
```

File 465 **lwarp-stabular.sty**

§ 574 Package stabular

(*Emulates or patches code by* Sigitas Tolušis.)

stabular (*Pkg*) stabular is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{stabular}[2014/03/20]
```

Env stabular [⟨*vpos*⟩] {⟨*colspec*⟩}

```
\newenvironment{stabular}[2][c]
{
\begin{tabular}[#1]{#2}
```

```
\renewcommand{\noalign}[1]{}
}
{\end{tabular}}
```

Env stabular {⟨*width*⟩} [⟨*vpos*⟩] {⟨*colspec*⟩}

```
\NewDocumentEnvironment{stabular*}{m o m}
{
\begin{tabular}[#2]{#3}
\renewcommand{\noalign}[1]{}
}
{\end{tabular}}
```

File 466 **lwarp-stackengine.sty**

§ 575 Package

stackengine

(*Emulates or patches code by* Steven B. Segletes.)

stackengine (*Pkg*) stackengine is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{stackengine}[2017/02/13]
```

The original version is neccessary for the patched \@stack and \stackanchor, where nesting lateximages does not work:

```
\LetLtxMacro\LWR@orig@stackengine\stackengine

\renewcommand*{\stackengine}[8]{%
    \ifstrequal{#4}{O}%
        {\begin{lateximage}[\ImageAltText]}%
        {\begin{lateximage}[\ImageAltText][][vertical-align:top]}%
    \LWR@orig@stackengine{#1}{#2}{#3}{#4}{#5}{#6}{#7}{#8}%
    \end{lateximage}%
}
```

\@stack uses a lateximage with a vertical alignment:

```
\LetLtxMacro\LWR@orig@@stack\@stack

\xpatchcmd{\LWR@orig@@stack}{\stackengine}{\LWR@orig@stackengine}
    {}
    {\LWR@patcherror{stackengine}{LWR@orig@@stack}}

\renewcommand*{\@stack}[4]{%
    \ifstrequal{#3}{O}%
        {\begin{lateximage}[\ImageAltText]}%
        {\begin{lateximage}[\ImageAltText][][vertical-align:top]}%
    \LWR@orig@@stack{#1}{#2}{#3}{#4}%
    \end{lateximage}%
}
```

The lapping macros are disabled for HTML:

```
\newcommand*\LWR@HTML@@stacklap[4]{#3}
\LWR@formatted{@stacklap}
```

\stackanchor is patched for two instances of \stackengine. A lateximage with vertical alignment is used.

```
\xpatchcmd{\stackanchor}{\stackengine}{\LWR@orig@stackengine}
    {}
    {\LWR@patcherror{stackengine}{stackanchor patch 1}}

\xpatchcmd{\stackanchor}{\stackengine}{\LWR@orig@stackengine}
    {}
    {\LWR@patcherror{stackengine}{stackanchor patch 2}}

\xpretocmd{\stackanchor}
    {\begin{lateximage}[\ImageAltText][][vertical-align:middle]}
    {}
    {\LWR@patcherror{stackengine}{stackanchor pre}}

\xapptocmd{\stackanchor}{\end{lateximage}}
    {}
    {\LWR@patcherror{stackengine}{stackanchor app}}
```

\Centerstack is simply placed inside a lateximage with a vertical alignment:

```
\xpretocmd{\Centerstack}
    {\begin{lateximage}[\ImageAltText][][vertical-align:middle]}
    {}
    {\LWR@patcherror{stackengine}{Centerstack pre}}

\xapptocmd{\Centerstack}{\end{lateximage}}
    {}
    {\LWR@patcherror{stackengine}{Centerstack app}}
```

\savestack reverts to print mode while saving the box, then places it inside a lateximage when used:

```
\renewcommand*\savestack[2]{%
  \xdef\sv@name{\stack@macro@name{#1}}%
  \@ifundefined{\sv@name content}{%
    \expandafter\newsavebox\expandafter{\csname\sv@name content\endcsname}%
   }{}%
    \begingroup%    lwarp
    \LWR@restoreorigformatting%     lwarp
   \RenewDocumentEnvironment{lateximage}{s o s o o d()}{}{}% lwarp: inside group
  \expandafter\LWR@gsavebox\csname\sv@name content\endcsname{#2}%
  \expandafter\gdef\expandafter#1\expandafter{%
        \expandafter\begin\expandafter{lateximage\expandafter}%     lwarp
        \expandafter\usebox\expandafter%
        {\csname\sv@name content\endcsname}%
        \expandafter\end\expandafter{lateximage\expandafter}%       lwarp
    }%
    \endgroup%      lwarp
}
```

File 467 **lwarp-stackrel.sty**

§ 576 Package stackrel

(*Emulates or patches code by* Heiko Oberdiek.)

stackrel (*Pkg*) stackrel is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{stackrel}[2016/05/16]

\begin{warpMathJax}
\CustomizeMathJax{\renewcommand{\stackrel}[3][]{%
    \mathrel{\mathop{#3}\limits_{#1}^{#2}}%
}}

\CustomizeMathJax{\newcommand{\stackbin}[3][]{%
    \mathbin{\mathop{#3}\limits_{#1}^{#2}}%
}}
\end{warpMathJax}
```

File 468 **lwarp-statex2.sty**

§ 577 Package statex2

(*Emulates or patches code by* Rodney A Sparapani.)

statex2 (*Pkg*) statex2 is patched for use by lwarp, and emulated for MathJax.

⚠ As of this version, option autobold does not appear to work for PDF output.

⚠ For MathJax, the tilde character ~ does not create \sim. Use \sim directly.

⚠ Because MathJax has limited conditional processing:

- \wrap only creates square braces, no matter what its optional arguments.
- \P, \pCau, \pN, and \pU do not handle special cases.

⚠ \and To have \and work if using \maketitle, place the following after the start of the document:

```
\newcommand*{\and}{%
    \relax\ifmmode%
        \expandafter\;\mb{\mathrm{and}}\;%
    \else%
        \expandafter\STATEXand%
    \fi%
}
```

for HTML output:

```
\LWR@ProvidesPackagePass{statex2}[2011/09/14]

\newcommand*{\LWR@HTML@Alpha}[1][]{%
    \fcolorbox{black}{ForestGreen}{\textcolor{white}{\textsf{ALPHA}}}%
    \textbf{\textcolor{ForestGreen}{\textsf{#1}}}%
}
\LWR@formatted{Alpha}

\newcommand*{\LWR@HTML@List}[1]{%
    \textbf{\textcolor{Dandelion}{\textsf{L}\textsubscript{\textit{#1}}}}%
}
```

```
\LWR@formatted{List}

\newcommand*{\LWR@HTML@Snd}[1][]{%
    \fcolorbox{black}{Dandelion}{\textcolor{white}{\textsf{2nd}}}%
    \textbf{\textcolor{Dandelion}{\textsf{#1}}}%
}
\LWR@formatted{Snd}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{statex2}

\CustomizeMathJax{\newcommand{\cpi}{\boldsymbol{\pi}}}
\CustomizeMathJax{\newcommand{\c}[1]{\boldsymbol{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\sfsl}[1]{\mathsf#1}}%     not slanted

\if@manualbold
\CustomizeMathJax{\newcommand{\mb}[1]{#1}}
\else
\CustomizeMathJax{\newcommand{\mb}[1]{\boldsymbol{#1}}}
\fi

\CustomizeMathJax{\newcommand{\diag}{\mb{\mathrm{diag}}}}
\CustomizeMathJax{\newcommand{\blockdiag}{\mb{\mathrm{blockdiag}}}}
\CustomizeMathJax{\newcommand{\erf}{\mb{\mathrm{erf}}}}
\CustomizeMathJax{\newcommand{\logit}{\mb{\mathrm{logit}}}}
\CustomizeMathJax{\newcommand{\trace}{\mb{\mathrm{trace}}}}

\CustomizeMathJax{\newcommand{\chisq}{{\mb{\chi^2}}}}
\CustomizeMathJax{\newcommand{\deriv}[2]{\mb{\frac{\d{}}{\d{#1}}}\wrap{\mb{#2}}}}
\CustomizeMathJax{\newcommand{\derivf}[2]{\mb{\frac{\d{}}{\d{#2}}}\wrap{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\e}[1]{\mb{\mathrm{e}^{#1}}}}
\CustomizeMathJax{\newcommand{\E}[2][]{\mb{\mathrm{E}}_{\mb{#1}} \wrap{\mb{#2}}}}
\CustomizeMathJax{\newcommand{\ha}{{\mb{\frac{\alpha}{2}}}}}
\CustomizeMathJax{\newcommand{\I}[2][]{%
    \mb{\mathrm{I}}_{\mb{#1}} \LWRwrapparen{\mb{#2}}%
}}
\CustomizeMathJax{\newcommand{\IBeta}[2]{%
    \mb{\frac{\Gamma[#1+#2]}{\Gamma[#1]\Gamma[#2]}}%
}}
\CustomizeMathJax{\newcommand{\If}{\;\mb{\mathrm{if}}\;}}
\CustomizeMathJax{\newcommand{\im}{\mb{\mathrm{i}}}}
\CustomizeMathJax{\newcommand{\ol}{\overline}}
\CustomizeMathJax{\newcommand{\ow}{\;\mb{\mathrm{otherwise}}\;}}
\CustomizeMathJax{\newcommand{\pderiv}[2]{%
    \mb{\frac{\partial}{\partial #1}}\wrap{\mb{#2}}%
}}
\CustomizeMathJax{\newcommand{\pderivf}[2]{%
    \mb{\frac{\partial}{\partial #2}}\wrap{\mb{#1}}%
}}
\CustomizeMathJax{\newcommand{\sd}{\mb{\sigma}}}
\CustomizeMathJax{\newcommand{\ul}{\underline}}
\CustomizeMathJax{\newcommand{\V}[2][]{\mb{\mathrm{V}}_{\mb{#1}} \wrap{\mb{#2}}}}
\CustomizeMathJax{\newcommand{\vs}{\;\mb{\mathrm{vs.}}\;}}
\CustomizeMathJax{\newcommand{\where}{\;\mb{\mathrm{where}}\;}}
\CustomizeMathJax{\newcommand{\wrap}[2][]{\left[ #2 \right]}}%       only []
\CustomizeMathJax{\newcommand{\LWRwrapparen}[1]{\left( #1 \right)}}%   lwarp

% \CustomizeMathJax{\renewcommand{~}{\mb{\sim}}}% doesn't work,
% replace <space>~<space> with <space>\sim<space>

```

```
\CustomizeMathJax{\newcommand{\iid}{\;\stackrel{\mb{\mathrm{iid}}}{\sim}\;}}
\CustomizeMathJax{\newcommand{\ind}{\;\stackrel{\mb{\mathrm{ind}}}{\sim}\;}}
\CustomizeMathJax{\newcommand{\indpr}{%
    \;\stackrel{\mb{\mathrm{ind}}}{\stackrel{\mb{\mathrm{prior}}}{\sim}}\;
}}
\CustomizeMathJax{\newcommand{\post}{\;\stackrel{\mb{\mathrm{post}}}{\sim}\;}}
\CustomizeMathJax{\newcommand{\prior}{\;\stackrel{\mb{\mathrm{prior}}}{\sim}\;}}

\CustomizeMathJax{\let\STATEXGamma=\Gamma}
\CustomizeMathJax{\renewcommand{\Gamma}[1][]{\mb{\STATEXGamma}\LWRwrapparen{\mb{#1}}}}
%
\CustomizeMathJax{\renewcommand{\and}{\;\mb{\mathrm{and}}\;}}
%
\CustomizeMathJax{\newcommand{\H}{\mb{\mathrm{H}}}}
%
\CustomizeMathJax{\newcommand{\P}[2][]{\mb{\mathrm{P}}_{\mb{#1}}\wrap{\mb{#2}}}}
%
\CustomizeMathJax{\newcommand{\|}{\mb{\mid}}}

\CustomizeMathJax{\newcommand{\B}[1]{\mb{\mathrm{B}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\BB}[1]{\mb{\mathrm{BetaBin}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\Bin}[2]{\mb{\mathrm{Bin}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\newcommand{\Dir}[1]{\mb{\mathrm{Dirichlet}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\HG}[3]{%
    \mb{\mathrm{Hypergeometric}}\LWRwrapparen{\mb{#1,\ #2,\ #3}}%
}}
\CustomizeMathJax{\newcommand{\M}[2]{%
    \mb{\mathrm{Multinomial}}\LWRwrapparen{\mb{#1,\ #2}}%
}}
\CustomizeMathJax{\newcommand{\NB}[2]{\mb{\mathrm{NegBin}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\newcommand{\Poi}[1]{\mb{\mathrm{Poisson}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\let\Poisson=\Poi}

\CustomizeMathJax{\newcommand{\pBB}[4][x]{%
    \mb{\frac{\Gamma[#2+1]\Gamma[#3+#1]\Gamma[#2+#4-#1]\Gamma[#3+#4]}%
    {\Gamma[#1+1]\Gamma[#2-#1+1]\Gamma[#2+#3+#4]\Gamma[#3]\Gamma[#4]}%
    \I[#1]{\{0, 1,\., #2\}}, \where #3>0,\; #4>0 \and n=1, 2,\.}%
}}
\CustomizeMathJax{\newcommand{\pBin}[3][x]{%
    \mb{\binom{#2}{#1}#3^{#1}} \LWRwrapparen{\mb{{1-#3}^{#2-#1}}}%
    \mb{\I[#1]{\{0,1,\.,#2\}}, \where p \in (0, 1) \and n=1, 2,\.}%
}}
\CustomizeMathJax{\newcommand{\pPoi}[2][x]{%
    \mb{\frac{1}{#1!}#2^{#1}\e{-#2}\I[#1]{\{0, 1,\.\}}, \where #2>0}%
}}

\CustomizeMathJax{\newcommand{\Cau}[2]{\mb{\mathrm{Cauchy}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\let\Cauchy=\Cau}
\CustomizeMathJax{\newcommand{\Chi}[2][]{%
    \chisq_{\mb{#1}}\LWRwrapparen{\mb{#2}}%
}}
\CustomizeMathJax{\let\Chisq=\Chi}
\CustomizeMathJax{\newcommand{\Bet}[2]{\mb{\mathrm{Beta}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\let\Beta=\Bet}
\CustomizeMathJax{\newcommand{\Exp}[1]{\mb{\mathrm{Exp}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\F}[2]{\mb{\mathrm{F}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\newcommand{\Gam}[2]{\mb{\mathrm{Gamma}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\newcommand{\IC}[1]{\mb{\mathrm{\chi^{-2}}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\IG}[2]{%
    \mb{\mathrm{Gamma^{-1}}}\LWRwrapparen{\mb{#1,\ #2}}%
```

```
}}
\CustomizeMathJax{\newcommand{\IW}[2]{%
    \mb{\mathrm{Wishart^{-1}}}\LWRwrapparen{\mb{#1,\ #2}}%
}}
\CustomizeMathJax{\newcommand{\Log}[2]{%
\mb{\mathrm{Logistic}}\LWRwrapparen{\mb{#1,\ #2}}%
}}
\CustomizeMathJax{\newcommand{\LogN}[2]{%
    \mb{\mathrm{Log\!-\!N}}\LWRwrapparen{\mb{#1,\ #2}}%
}}
\CustomizeMathJax{\newcommand{\N}[3][]{%
    \mb{\mathrm{N}}_{\mb{#1}}\LWRwrapparen{\mb{#2,\ #3}}%
}}
\CustomizeMathJax{\newcommand{\Par}[2]{\mb{\mathrm{Pareto}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\let\Pareto=\Par}
\CustomizeMathJax{\newcommand{\Tsq}[2]{\mb{\mathrm{T^2}}\LWRwrapparen{\mb{#1,\ #2}}}}
\CustomizeMathJax{\newcommand{\U}[1]{\mb{\mathrm{U}}\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\W}[2]{\mb{\mathrm{Wishart}}\LWRwrapparen{\mb{#1,\ #2}}}}

\CustomizeMathJax{\renewcommand{\t}[1]{\mb{\mathrm{t}}\LWRwrapparen{\mb{#1}}}}

\CustomizeMathJax{\newcommand{\pBet}[3][x]{%
    \IBeta{#2}{#3}%
    #1^{#2-1}\LWRwrapparen{1-#1}^{#3-1}\I[#1]{0,\ 1}, \where #2>0 \and #3>0%
}}
\CustomizeMathJax{\newcommand{\pCau}[3][x]{%
%    \ifthenelse{\equal{#2, #3}{0, 1}}{\frac{1}{\cpi\LWRwrapparen{1+#1}^2}}%
  {\frac{1}{#3\cpi\left\{1+\wrap{\LWRwrapparen{x-#2}/#3}^2\right\}}, \where #3>0}%
}}% no special case for 0,1
\CustomizeMathJax{\newcommand{\pChi}[2][x]{%
    \frac{2^{-#2/2}}{\Gamma[#2/2]}#1^{#2/2-1}\e{-#1/2}%
    \I[#1]{0,\infty}, \where #2>0%
}}
\CustomizeMathJax{\newcommand{\pExp}[2][x]{%
    \frac{1}{#2}\e{-#1/#2}\I[#1]{0,\infty},%
    \where #2>0%
}}
\CustomizeMathJax{\newcommand{\pGam}[3][x]{%
    \frac{#3^{#2}}{\Gamma[#2]}#1^{#2-1}\e{-#3#1}%
    \I[#1]{0,\infty}, \where #2>0 \and #3>0%
}}
\CustomizeMathJax{\newcommand{\pN}[3][x]{%
%     \ifthenelse{\equal{#2, #3}{0, 1}}%
%     {\frac{1}{\sqrt{2\cpi}}\e{-#1^2/2}}%
    {\frac{1}{\sqrt{2\cpi \cdot #3}}\e{-\LWRwrapparen{#1-#2}^2/2 \cdot #3}}%
}}% no test for 0,1, must add \cdot
\CustomizeMathJax{\newcommand{\pPar}[3][x]{%
    \frac{#3}{#2\LWRwrapparen{1+#1/#2}^{#3+1}}\I[#1]{0,\infty},%
    \where #2>0 \and #3>0%
}}
\CustomizeMathJax{\newcommand{\pU}[3][x]{%
%     \ifthenelse{\equal{#2, #3}{0, 1}}{\I[#1]{0,\ 1}}%
    {\frac{1}{#3-#2}\I[#1]{#2,\ #3}, \where #2<#3}%
}}% no special case for 0,1

\CustomizeMathJax{\newcommand{\=}[1]{\bar{#1}}}
\CustomizeMathJax{\let\^\widehat}
\CustomizeMathJax{\let\~\widetilde}
\CustomizeMathJax{\newcommand{\'}[1]{\LWRwrapparen{\mb{#1}}}}
\CustomizeMathJax{\newcommand{\b}[1]{\bar{#1}}}
```

```
\CustomizeMathJax{\newcommand{\c}[1]{\mb{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\d}[1]{\,\mb{\mathrm{d}{#1}}}}
\CustomizeMathJax{\newcommand{\.}{\mb{\ldots}}}
\end{warpMathJax}
```

File 469 **lwarp-statistics.sty**

§ 578 Package

statistics

(*Emulates or patches code by* Julien Rivaud.)

statistics (*Pkg*) statistics is patched for use by lwarp.

⚠ \color The statistics documentation examples include the use of the \color macro. Use \textcolor instead.

⚠ math The statistics package uses math arrays, but the HTML version uses text tabulars to allow text copy/paste. If math is required, use \ensuremath or \(and \) as needed.

Pre/postline is ignored, and \hline is used instead. Each table will have an \hline above and below as a frame.

for HTML output:

```
\LWR@ProvidesPackagePass{statistics}[2019/09/29]
```

```
\ExplSyntaxOn
```

To use text tabular instead of math array. This allows text copy/paste of the results.

In the following, all changes for the Lwarp package are labelled "lwarp".

Redefined using the lwarp version of &:

```
\StartDefiningTabulars%     lwarp
\cs_set_protected_nopar:Nn \__statistics_table_make:nn {
    \int_compare:nT
            { 0 < \l__statistics_table_maxcols_int
                = \l__statistics_nbvals_int } {
        \__statistics_table_end:
        \tl_use:N \l__statistics_table_sep_tl
        \__statistics_table_start:
    }
    \int_incr:N \l__statistics_nbvals_int
    \int_incr:N \l__statistics_currange_int
    \fp_add:Nn \l__statistics_curtotal_fp { #2 }
    \__statistics_set_if_shown:N \l_tmpa_bool
    \tl_set:Nx \l_tmpa_tl {
        \exp_not:n { & \tl_set:Nn \currentcolumn } {
            \int_use:N \l__statistics_currange_int
        }
    }
    \bool_if:NTF \l_tmpa_bool {
        \tl_put_right:Nn \l_tmpa_tl
            {\__statistics_table_shown_format:n}
    }{
        \tl_put_right:Nn \l_tmpa_tl
```

```
            {\__statistics_table_hidden_format:n}
    }
    \seq_put_right:Nn \l__statistics_store_values_seq { #1 }
    \bool_if:NT \l__statistics_table_values_bool {
        \tl_put_right:Nx \l__statistics_table_values_tl {
            \exp_not:V \l_tmpa_tl {
                \exp_not:n {
                    \__statistics_table_values_format:n { #1 }
                }
            }
        }
    }
    \seq_put_right:Nx \l__statistics_store_counts_seq { \fp_eval:n {#2} }
    \bool_if:NT \l__statistics_table_counts_bool {
        \tl_put_right:Nx \l__statistics_table_counts_tl {
            \exp_not:V \l_tmpa_tl {
                \exp_not:n {
                    \__statistics_table_counts_format:n {
                        { \__statistics_table_allcounts_format:n { #2 } }
                    }
                }
            }
        }
    }
    \bool_if:NT \l__statistics_table_icc_bool {
        \tl_put_right:Nx \l__statistics_table_icc_tl {
            \exp_not:V \l_tmpa_tl {
                \exp_not:n { \__statistics_table_icc_format:n }
                {
                    \exp_not:n{ \__statistics_table_allcounts_format:n }
                        { \fp_use:N \l__statistics_curtotal_fp }
                }
            }
        }
    }
    \bool_if:NT \l__statistics_table_dcc_bool {
        \tl_put_right:Nx \l__statistics_table_dcc_tl {
            \exp_not:V \l_tmpa_tl {
                \exp_not:n { \__statistics_table_dcc_format:n }
                {
                    \exp_not:n{ \__statistics_table_allcounts_format:n }
                    {
                        \fp_eval:n {
                            \l__statistics_total_fp
                                - \l__statistics_curtotal_fp
                                + #2
                        }
                    }
                }
            }
        }
    }
    \fp_set:Nn \l__statistics_table_curICF_fp {
        round(\l__statistics_curtotal_fp
                / \l__statistics_total_fp,
              \l__statistics_table_round_int)
    }
    \bool_if:NT \l__statistics_table_frequencies_bool {
        \tl_put_right:Nx \l__statistics_table_frequencies_tl {
            \exp_not:V \l_tmpa_tl {
```

```
                \exp_not:n { \__statistics_table_frequencies_format:n }
                {
                    \exp_not:n{ \__statistics_table_allfreqs_format:n }
                    {
                        \fp_eval:n {
                            \l__statistics_table_curICF_fp
                                - \l__statistics_table_prevICF_fp
                        }
                    }
                }
            }
        }
    }
    \bool_if:NT \l__statistics_table_icf_bool {
        \tl_put_right:Nx \l__statistics_table_icf_tl {
            \exp_not:V \l_tmpa_tl {
                \exp_not:n { \__statistics_table_icf_format:n }
                {
                    \exp_not:n{ \__statistics_table_allfreqs_format:n }
                        { \fp_to_decimal:N \l__statistics_table_curICF_fp }
                }
            }
        }
    }
    \bool_if:NT \l__statistics_table_dcf_bool {
        \tl_put_right:Nx \l__statistics_table_dcf_tl {
            \exp_not:V \l_tmpa_tl {
                \exp_not:n { \__statistics_table_dcf_format:n }
                {
                    \exp_not:n{ \__statistics_table_allfreqs_format:n }
                    {
                        \fp_eval:n {
                            1 - \l__statistics_table_prevICF_fp
                        }
                    }
                }
            }
        }
    }
    \fp_set_eq:NN
        \l__statistics_table_prevICF_fp
        \l__statistics_table_curICF_fp
}
\StopDefiningTabulars%  lwarp
```

Redefined using `tabular`. Also, preline and postline do not work correctly with lwarp, which looks for certain tokens to detect `\hlines`, so `\hline` is used instead.

```
\cs_set_protected_nopar:Nn \__statistics_table_end: {
    \tl_set:Nx \l__statistics_table_preamble_tl {
%         \exp_not:n { \begin{array}[ }
        \exp_not:n {\begin{tabular}[ }%      lwarp
            \exp_not:V \l__statistics_table_valign_tl
        \exp_not:n { ] }
                { \exp_not:V \l__statistics_table_headcoltype_tl
                  \prg_replicate:nn { \l__statistics_nbvals_int }
                    { \exp_not:V \l__statistics_table_coltype_tl } }
    }
    \seq_clear:N \l__statistics_table_contents_seq
   \clist_map_inline:nn { values, counts, icc, dcc, frequencies, icf, dcf } {
```

```
        \bool_if:cT { l__statistics_table_##1_bool } {
            \seq_put_right:Nv
                    \l__statistics_table_contents_seq
                    { l__statistics_table_##1_tl }
        }
    }
%     $
    \tl_use:N \l__statistics_table_preamble_tl
        \hline%      lwarp
%         \l__statistics_table_preline_tl
        \seq_use:Nn
                \l__statistics_table_contents_seq
                { \l__statistics_table_newline_tl }
        \\
%         \l__statistics_table_postline_tl
        \hline%      lwarp
%     \end{array}$
    \end{tabular}%      lwarp
}
```

With lwarp, `\ensuremath` creates an SVG image, but its `alt` tag does not contain the text of the contents for copy/paste, since these expressions are usually not simple text. For the statistics package, copy/paste is restored by using text instead of math output.

For the leftmost column. Redefined to use text output:

```
\cs_set_protected_nopar:Nn \__statistics_table_start: {
    \int_zero:N \l__statistics_nbvals_int
    \clist_pop:NNT \l__statistics_table_maxcols_clist \l_tmpa_tl {
        \int_set:Nn \l__statistics_table_maxcols_int { \l_tmpa_tl }
    }
   \clist_map_inline:nn { values, counts, frequencies, icc, icf, dcc, dcf } {
        \tl_set:cx { l__statistics_table_##1_tl } {
%             \exp_not:N \ensuremath {
                \exp_not:N \hbox {
                    \exp_not:c { l__statistics_table_##1_name_tl }
                }
%             }
        }
    }
}
```

For the first row. Redefined to use text output:

```
\RenewDocumentCommand \__statistics_IN:w { m u{;} u{;} m } {
%     \ensuremath{ \left#1 \num{#2} \mathbin{;} \num{#3} \right#4 }
    #1 #2 ; #3 #4%      lwarp
}

\__statistics_setup:nn { table } {
%     values/format = \ensuremath{#1},
    values/format = {#1},%      lwarp
}
```

Added `\ExplSyntaxOn/Off` to avoid errors. (In once instance, a double subscript error appeared.)

```
\RenewDocumentCommand \StatsGraph { +O{} +m +O{} } {
```

```
    \group_begin:
    \int_gincr:N \g__statistics_graph_last_int
    \tl_set:Nx \l_tmpa_tl {
        \exp_not:n { g__statistics_graph_xstep_ }
        \int_use:N \g__statistics_graph_last_int
        \exp_not:n { _tl }
    }
    \tl_if_exist:cTF { \l_tmpa_tl } {
        \fp_gset:Nn \g__statistics_graph_xstep_fp
            { \tl_use:c {\l_tmpa_tl} }
    }{
        \fp_gset:Nn \g__statistics_graph_xstep_fp { \c_one_int }
    }
    \__statistics_setup:nn { graph } { #1, #3 }
    \tl_if_single:nTF { #2 } {
        \cs_if_exist:NF #2 { #2 }
        \tl_set_eq:NN \l__statistics_data_tl #2
    }{
        \tl_set:Nn \l__statistics_data_tl { #2 }
    }
    \fp_zero:N \l__statistics_graph_maxheight_fp
    \fp_set:Nn \l__statistics_graph_minvalue_fp {inf}
    \fp_set:Nn \l__statistics_graph_maxvalue_fp {-inf}
    \fp_zero:N \l__statistics_total_fp
    \int_zero:N \l__statistics_nbvals_int
    \bool_set_true:N \l__statistics_graph_allranges_bool
    \keyval_parse:NNV
            \__statistics_graph_prepare:n
            \__statistics_graph_prepare:nn
            \l__statistics_data_tl
    \tl_clear:N \l__statistics_graph_tikzdata_tl
    \tl_clear:N \l__statistics_graph_tikzinfo_tl
    \int_zero:N \l__statistics_currange_int
    \bool_if:NTF \l__statistics_graph_allranges_bool {
        \bool_if:NTF \l__statistics_graph_cumulative_bool {
\ExplSyntaxOn%     lwarp
            \__statistics_graph_dopicture_cumulative:
\ExplSyntaxOff%    lwarp
       }{
\ExplSyntaxOn%     lwarp
            \__statistics_graph_dopicture_hist:
\ExplSyntaxOff%    lwarp
       }
   }{
\ExplSyntaxOn%    lwarp
       \__statistics_graph_dopicture_comb:
\ExplSyntaxOff%    lwarp
   }
   \iow_now:Nx \@auxout {
       \exp_not:n {
           \ExplSyntaxOn
           \tl_gset:cn
       }
       {
           \exp_not:n {g__statistics_graph_xstep_}
           \int_use:N \g__statistics_graph_last_int
           \exp_not:n {_tl}
       }
       {
           \fp_to_decimal:N \g__statistics_graph_xstep_fp
```

```
        }
        \exp_not:n {
            \ExplSyntaxOff
        }
    }
    \group_end:
}

\ExplSyntaxOff
```

File 470 **lwarp-statmath.sty**

§ 579 Package statmath

(*Emulates or patches code by* Sebastian Ankargren.)

statmath (*Pkg*) statmath is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{statmath}[2018/03/08]

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{statmath}

\ifdefequal{\abcbf}{\mathbf}
    {\CustomizeMathJax{\newcommand{\abcbf}[1]{\mathbf{#1}}}}
    {\CustomizeMathJax{\newcommand{\abcbf}[1]{\boldsymbol{#1}}}}

\CustomizeMathJax{\newcommand{\greekbf}[1]{\boldsymbol{#1}}}

\CustomizeMathJax{\newcommand{\bfA}{\abcbf A}}
\CustomizeMathJax{\newcommand{\bfB}{\abcbf B}}
\CustomizeMathJax{\newcommand{\bfC}{\abcbf C}}
\CustomizeMathJax{\newcommand{\bfD}{\abcbf D}}
\CustomizeMathJax{\newcommand{\bfE}{\abcbf E}}
\CustomizeMathJax{\newcommand{\bfF}{\abcbf F}}
\CustomizeMathJax{\newcommand{\bfG}{\abcbf G}}
\CustomizeMathJax{\newcommand{\bfH}{\abcbf H}}
\CustomizeMathJax{\newcommand{\bfI}{\abcbf I}}
\CustomizeMathJax{\newcommand{\bfJ}{\abcbf J}}
\CustomizeMathJax{\newcommand{\bfK}{\abcbf K}}
\CustomizeMathJax{\newcommand{\bfL}{\abcbf L}}
\CustomizeMathJax{\newcommand{\bfM}{\abcbf M}}
\CustomizeMathJax{\newcommand{\bfN}{\abcbf N}}
\CustomizeMathJax{\newcommand{\bfO}{\abcbf O}}
\CustomizeMathJax{\newcommand{\bfP}{\abcbf P}}
\CustomizeMathJax{\newcommand{\bfQ}{\abcbf Q}}
\CustomizeMathJax{\newcommand{\bfR}{\abcbf R}}
\CustomizeMathJax{\newcommand{\bfS}{\abcbf S}}
\CustomizeMathJax{\newcommand{\bfT}{\abcbf T}}
\CustomizeMathJax{\newcommand{\bfU}{\abcbf U}}
\CustomizeMathJax{\newcommand{\bfV}{\abcbf V}}
\CustomizeMathJax{\newcommand{\bfW}{\abcbf W}}
\CustomizeMathJax{\newcommand{\bfX}{\abcbf X}}
\CustomizeMathJax{\newcommand{\bfY}{\abcbf Y}}
\CustomizeMathJax{\newcommand{\bfZ}{\abcbf Z}}
```

```
\CustomizeMathJax{\newcommand{\bfa}{\abcbf a}}
\CustomizeMathJax{\newcommand{\bfb}{\abcbf b}}
\CustomizeMathJax{\newcommand{\bfc}{\abcbf c}}
\CustomizeMathJax{\newcommand{\bfd}{\abcbf d}}
\CustomizeMathJax{\newcommand{\bfe}{\abcbf e}}
\CustomizeMathJax{\newcommand{\bff}{\abcbf f}}
\CustomizeMathJax{\newcommand{\bfg}{\abcbf g}}
\CustomizeMathJax{\newcommand{\bfh}{\abcbf h}}
\CustomizeMathJax{\newcommand{\bfi}{\abcbf i}}
\CustomizeMathJax{\newcommand{\bfj}{\abcbf j}}
\CustomizeMathJax{\newcommand{\bfk}{\abcbf k}}
\CustomizeMathJax{\newcommand{\bfl}{\abcbf l}}
\CustomizeMathJax{\newcommand{\bfm}{\abcbf m}}
\CustomizeMathJax{\newcommand{\bfn}{\abcbf n}}
\CustomizeMathJax{\newcommand{\bfo}{\abcbf o}}
\CustomizeMathJax{\newcommand{\bfp}{\abcbf p}}
\CustomizeMathJax{\newcommand{\bfq}{\abcbf q}}
\CustomizeMathJax{\newcommand{\bfr}{\abcbf r}}
\CustomizeMathJax{\newcommand{\bfs}{\abcbf s}}
\CustomizeMathJax{\newcommand{\bft}{\abcbf t}}
\CustomizeMathJax{\newcommand{\bfu}{\abcbf u}}
\CustomizeMathJax{\newcommand{\bfv}{\abcbf v}}
\CustomizeMathJax{\newcommand{\bfw}{\abcbf w}}
\CustomizeMathJax{\newcommand{\bfx}{\abcbf x}}
\CustomizeMathJax{\newcommand{\bfy}{\abcbf y}}
\CustomizeMathJax{\newcommand{\bfz}{\abcbf z}}

\LWR@mathjax@addgreek@l@bfit{bf}{}% Greek lowercase bold face italic
\LWR@mathjax@addgreek@u@bfup*{bf}{}% Greek uppercase bold face upright, cap macros.

\CustomizeMathJax{\newcommand{\bfzero}{\greekbf 0}}

\CustomizeMathJax{\DeclareMathOperator{\cov}{Cov}}
\CustomizeMathJax{\DeclareMathOperator{\E}{E}}
\CustomizeMathJax{\DeclareMathOperator{\V}{V}}
\CustomizeMathJax{\newcommand{\inas}{\overset{a.s.}{\to}}}
\CustomizeMathJax{\newcommand{\indist}{\overset{d}{\to}}}
\CustomizeMathJax{\newcommand{\inprob}{\overset{p}{\to}}}
\CustomizeMathJax{\DeclareMathOperator{\plim}{plim}}
\CustomizeMathJax{\DeclareMathOperator{\tr}{tr}}
\CustomizeMathJax{\DeclareMathOperator{\vc}{vec}}
\CustomizeMathJax{\DeclareMathOperator{\vcs}{vecs}}
\CustomizeMathJax{\DeclareMathOperator{\vch}{vech}}
\CustomizeMathJax{\DeclareMathOperator{\diag}{diag}}
\CustomizeMathJax{\DeclareMathOperator{\argmin}{arg\,min}}
\CustomizeMathJax{\DeclareMathOperator{\argmax}{arg\,max}}
\end{warpMathJax}
```

File 471 **lwarp-steinmetz.sty**

§ 580 Package steinmetz

(*Emulates or patches code by* Enrico Gregorio.)

steinmetz (*Pkg*) steinmetz is patched for use by lwarp. Emulation is provided for MathJax

for HTML output:

```
\LWR@ProvidesPackagePass{steinmetz}[2009/06/14]
```

```
\renewcommand{\phase}[2][]{%
    \begin{lateximage}*[steinmetz\{\detokenize{#2}\}]
    \ensuremath{\underline{/#2}}
    \end{lateximage}
}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\phase}[2][]{\underline{/#2}}}
\end{warpMathJax}
```

File 472 **lwarp-stfloats.sty**

§ 581 Package stfloats

`stfloats` (*Pkg*) stfloats is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{stfloats}[2017/03/27]
```

stfloats may have been preloaded by a ltj* class.

The following are provided in case they have not yet been defined:

```
\providecommand*{\fnbelowfloat}{}
\providecommand*{\fnunderfloat}{}
\providecommand*{\setbaselinefloat}{}
\providecommand*{\setbaselinefixed}{}
```

Nullified for HTML:

```
\renewcommand*{\fnbelowfloat}{}
\renewcommand*{\fnunderfloat}{}
\renewcommand*{\setbaselinefloat}{}
\renewcommand*{\setbaselinefixed}{}
```

File 473 **lwarp-struktex.sty**

§ 582 Package struktex

(*Emulates or patches code by* Jobst Hoffmann.)

`struktex` (*Pkg*) struktex is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{struktex}

\BeforeBeginEnvironment{struktogramm}{%
    \begin{lateximage}[-struktex-~\PackageDiagramAltText]%
}
\AfterEndEnvironment{struktogramm}{\end{lateximage}}

\newenvironment{LWR@HTML@centernss}{\begin{center}}{\end{center}}
\LWR@formattedenv{centernss}

\newcommand{\LWR@HTML@CenterNssFile}[1]{%
    \begin{center}
```

```
    \input{#1.nss}
    \end{center}
}
\LWR@formatted{CenterNssFile}

\newcommand{\LWR@HTML@centernssfile}{\LWR@HTML@CenterNssFile}
\LWR@formatted{centernssfile}
```

File 474 **lwarp-subcaption.sty**

§ 583 Package subcaption

(*Emulates or patches code by* Axel Sommerfeldt.)

subcaption (*Pkg*) subcaption is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{subcaption}[2018/05/01]
```

Tells lwarp to ignore minipage widths inside a subfigure or subtable. In print mode the minipages are used to place the items next to each other. In HTML they are placed side-by-side automatically.

```
\xpretocmd{\subcaption@iiminipage}
    {\minipagefullwidth}
    {}
    {\LWR@patcherror{subcaption}{subcaption@iiminipage}}
```

Likewise for a \subcaptionbox:

```
\xpretocmd{\subcaptionbox}
    {\minipagefullwidth}
    {}
    {\LWR@patcherror{subcaption}{subcaptionbox}}
```

File 475 **lwarp-subfig.sty**

§ 584 Package subfig

(*Emulates or patches code by* Steven Douglas Cochran.)

subfig (*Pkg*) subfig is supported and patched by lwarp.

⚠ table numbering To have correct sub table numbers:

```
\usepackage{caption}
\captionsetup[table]{position=top}
```

⚠ lof/lotdepth At present, the package options for lofdepth and lotdepth are not working. These counters must be set separately after the package has been loaded.

⚠ horizontal spacing In the document source, use \hfill and \hspace* between subfigures to spread them apart horizontally. The use of other forms of whitespace may cause paragraph tags to be generated, resulting in subfigures appearing on the following lines instead of all on a single line.

for HTML output: Accept all options for lwarp-subfig:

```
\LWR@ProvidesPackagePass{subfig}[2005/06/28]
```

\sf@@@subfloat {⟨*1 type*⟩} [⟨*2 lof entry*⟩] [⟨*3 caption*⟩] {⟨*4 contents*⟩}

The outer minipage allows side-by-side subfloats with \hfill between.

```
\long\def\sf@@@subfloat#1[#2][#3]#4{%
\begin{minipage}{\linewidth}% lwarp

\IfValueTF{#2}{%
    \LWR@setlatestname{#2}%
}{%
    \IfValueTF{#3}{%
        \LWR@setlatestname{#3}%
    }{}%
}%
\LWR@stoppars% lwarp
    \@ifundefined{FBsc@max}{}%
        {\FB@readaux{\let\FBsuboheight\relax}}%
    \@tempcnta=\@ne
    \if@minipage
      \@tempcnta=\z@
    \else\ifdim \lastskip=\z@ \else
      \@tempcnta=\tw@
    \fi\fi
    \ifmaincaptiontop
      \sf@top=\sf@nearskip
      \sf@bottom=\sf@farskip
    \else
      \sf@top=\sf@farskip
      \sf@bottom=\sf@nearskip
    \fi
    \leavevmode

%     \setbox\@tempboxa \hbox{#4}%
%     \@tempdima=\wd\@tempboxa
%     \@ifundefined{FBsc@max}{}%
%         {\global\advance\Xhsize-\wd\@tempboxa
%          \dimen@=\ht\@tempboxa
%          \advance\dimen@\dp\@tempboxa
%          \ifdim\dimen@>\FBso@max
%            \global\FBso@max\dimen@
%          \fi}%
```

Do not use boxes, which interfere with lateximages:

```
%     \vtop%
    \bgroup
%       \vbox%
      \bgroup
        \ifcase\@tempcnta
          \@minipagefalse
        \or
%           \vskip\sf@top
        \or
          \ifdim \lastskip=\z@ \else
%             \@tempskipb\sf@top\relax\@xaddvskip
          \fi
        \fi
```

```
        \sf@ifpositiontop{%
          \ifx \@empty#3\relax \else
            \sf@subcaption{#1}{#2}{#3}%
% \vskip\sf@capskip
% \vskip\sf@captopadj
          \fi\egroup
% \hrule width0pt height0pt depth0pt
          \LWR@startpars% lwarp
% \box\@tempboxa
          #4
          \LWR@stoppars% lwarp
        }{%
        \LWR@startpars% lwarp
        \@ifundefined{FBsc@max}%
            {
% \box\@tempboxa
               #4
            }%
            {\ifx\FBsuboheight\relax
% \box\@tempboxa
               #4
             \else
% \vbox to \FBsuboheight{\FBafil\box\@tempboxa\FBbfil}%
               #4
             \fi}%
        \LWR@stoppars% lwarp
          \egroup
          \ifx \@empty#3\relax \else
% \vskip\sf@capskip
% \hrule width0pt height0pt depth0pt
            \sf@subcaption{#1}{#2}{#3}%
         \fi
        }%
% \vskip\sf@bottom
    \egroup
    \@ifundefined{FBsc@max}{}%
        {\addtocounter{FRobj}{-1}%
         \ifnum\c@FRobj=0\else
           \subfloatrowsep
         \fi}%
    \ifmaincaptiontop\else
      \global\advance\@nameuse{c@\@captype}\m@ne
    \fi
\end{minipage}% lwarp
\LWR@startpars% lwarp
  \endgroup\ignorespaces%
}%
```

\sf@subcaption {⟨*1 type*⟩} {⟨*2 lof entry*⟩} {⟨*3 caption*⟩}

```
\long\def\sf@subcaption#1#2#3{%
\LWR@stoppars% lwarp
  \ifx \relax#2\relax \else
    \bgroup
      \let\label=\@gobble
      \let\protect=\string
      \def\@subcaplabel{%
        \caption@lstfmt{\@nameuse{p@#1}}{\@nameuse{the#1}}}%
      \sf@updatecaptionlist{#1}{#2}{\the\value{\@captype}}{\the\value{#1}}%
    \egroup
```

```
  \fi
  \bgroup
    \ifx \relax#3\relax
      \let\captionlabelsep=\relax
    \fi
%     \setbox0\vbox{%
%        \hb@xt@\the\@tempdima{%
%
% %          \hss
% %         \parbox[t]{\the\@tempdima}{%
%           \caption@make
%               {\@nameuse{sub\@captype name}}%
%               {\@nameuse{thesub\@captype}}%
%               {#3}
% %  }%
% %            \hss
%   }
%  }%
    \@ifundefined{FBsc@max}%
%         {\box0}%
        {
%  \parbox[t]{\the\@tempdima}{%
\LWR@traceinfo{sfsubcap B1}%    lwarp
            \LWR@figcaption%    lwarp
            \caption@make
                {\@nameuse{sub\@captype name}}%
                {\@nameuse{thesub\@captype}}%
                {\LWR@isolate{#3}}%

            \endLWR@figcaption%     lwarp
\LWR@traceinfo{sfsubcap B2}%        lwarp
%  }%
        }%
        {\dimen@\ht0%
         \advance\dimen@\dp0%
         \ifdim\dimen@>\FBsc@max
           \global\FBsc@max\dimen@
         \fi
         \FB@readaux{\let\FBsubcheight\relax}%
         \ifx\FBsubcheight\relax
           \def\next{
%  \parbox[t]{\the\@tempdima}
            }%
         \else
           \def\next{
%  \parbox[t][\FBsubcheight][t]{\the\@tempdima}
            }%
         \fi
%           \vbox{%
%             \hb@xt@\the\@tempdima{%
%
%               \hss
%               \next{%
\LWR@traceinfo{sfsubcap C1}% lwarp
               \caption@make
                   {\@nameuse{sub\@captype name}}%
                   {\@nameuse{thesub\@captype}}%
                   {#3}
\LWR@traceinfo{sfsubcap C1}% lwarp
%  }%
```

```
%               \hss
%
%  }
%           }
        }%
  \egroup
\LWR@startpars% lwarp
}
```

\subfloat@label

Patches for \sf@sub@label:

```
\xpretocmd{\subfloat@label}
    {\LWR@ensuredoingapar}
    {}
    {\LWR@patcherror{subfig}{subfloat@label}}
```

Patches for \subref.

\sf@subref

{⟨*label*⟩}

The unstarred version uses a \ref link whose printed text comes from the sub@<label>:

```
\renewcommand{\sf@subref}[1]{%
    \LWR@subnewref{#1}{sub@#1}%
}
```

\sf@@subref

{⟨*label*⟩}

The starred version uses the printed sub@<label> which is stored as if it were a page number:

```
\renewcommand{\sf@@subref}[1]{\LWR@orig@pageref{sub@#1}}
```

Defining new subfloats. The l@sub<type> for each is redefined.

\@newsubfloat

[⟨*keys/values*⟩] {⟨*float name*⟩}

```
\LetLtxMacro\LWR@orig@newsubfloat\@newsubfloat

\def\@newsubfloat[#1]#2{%
\LWR@orig@newsubfloat[#1]{#2}%
\renewcommand{\l@sub#2}[2]{\hypertocfloat{2}{sub#2}{\ext@sub#2}{##1}{##2}}%
}
```

Pre-defined for figures and tables:

\l@subfigure

{⟨*text*⟩} {⟨*pagenum*⟩}

```
\renewcommand{\l@subfigure}[2]{\hypertocfloat{2}{subfigure}{lof}{#1}{#2}}
```

\l@subtable

{⟨*text*⟩} {⟨*pagenum*⟩}

```
\renewcommand{\l@subtable}[2]{\hypertocfloat{2}{subtable}{lot}{#1}{#2}}
```

File 476 **lwarp-subfigure.sty**

§ 585 Package subfigure

subfigure (*Pkg*) subfigure is emulated by subfig.

for HTML output:

```
\LWR@ProvidesPackageDrop{subfigure}[2002/03/15]
\RequirePackage{subfig}

\LetLtxMacro\subfigure\subfloat
\LetLtxMacro\subtable\subfloat
\LetLtxMacro\Subref\subref
\@ifundefined{figuretopcaptrue}{\newif\iffiguretopcap}{}
\newif\ifsubfiguretopcap
\newif\ifsubcaphang
\newif\ifsubcapcenter
\newif\ifsubcapcenterlast
\newif\ifsubcapnooneline
\newif\ifsubcapraggedright
\newskip\subfigtopskip
\newskip\subfigcapskip
\newdimen\subfigcaptopadj
\newskip\subfigbottomskip
\newdimen\subfigcapmargin
\newskip\subfiglabelskip
\newcommand*{\subcapsize}{}
\newcommand*{\subcaplabelfont}{}
\newcommand*{\subcapfont}{}
```

File 477 **lwarp-subsupscripts.sty**

§ 586 Package subsupscripts

(*Emulates or patches code by* Riccardo Bresciani.)

subsupscripts (*Pkg*) subsupscripts is used as-is for SVG math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{subsupscripts}[2009/10/27]
```

The larger skips are used here.

```
\begin{warpMathJax}
\CustomizeMathJax{%
    \newcommand{\fourscriptsC}[7]{%
        {}^{#2}_{#3}\hspace{#6}#1\hspace{#7}{}^{#4}_{#5}%
    }
}
\CustomizeMathJax{%
    \newcommand{\lrsubscriptsC}[5]{%
        \fourscriptsC{#1}{}{#2}{}{#3}{#4}{#5}%
    }
}
```

```
\CustomizeMathJax{%
    \newcommand{\lrsuperscriptsC}[5]{%
        \fourscriptsC{#1}{#2}{}{#3}{}{#4}{#5}%
    }
}
\CustomizeMathJax{%
    \newcommand{\fourscripts}[5]{%
        \fourscriptsC{#1}{#2}{#3}{#4}{#5}{0ex}{0ex}%
    }
}
\CustomizeMathJax{%
    \newcommand{\lrsubscripts}[3]{\fourscripts{#1}{}{#2}{}{#3}}
}
\CustomizeMathJax{%
    \newcommand{\lrsuperscripts}[3]{\fourscripts{#1}{#2}{}{#3}{}}
}
\CustomizeMathJax{%
    \newcommand{\twolscripts}[4][-.16ex]{{}^{#3}_{#4}\hspace{#1}#2}
}
\CustomizeMathJax{%
    \newcommand{\tworscripts}[4][-.07ex]{#2\hspace{#1}{}^{#3}_{#4}}
}
\CustomizeMathJax{%
    \newcommand{\lsubscript}[3][-.16ex]{\twolscripts[#1]{#2}{}{#3}}
}
\CustomizeMathJax{%
    \newcommand{\lsuperscript}[3][-.16ex]{\twolscripts[#1]{#2}{#3}{}}
}
\CustomizeMathJax{%
    \newcommand{\rsubscript}[3][-.07ex]{\tworscripts[#1]{#2}{}{#3}}
}
\CustomizeMathJax{%
    \newcommand{\rsuperscript}[3][-.07ex]{\tworscripts[#1]{#2}{#3}{}}
}
\end{warpMathJax}
```

File 478 **lwarp-supertabular.sty**

§ 587 Package

supertabular

(*Emulates or patches code by* Johannes Braams, Theo Jurriens.)

`supertabular` (*Pkg*) supertabular is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{supertabular}[2004/02/20]
```

⚠ Misplaced alignment tab character &

For `\tablefirsthead`, etc., enclose them as follows:

```
\StartDefiningTabulars
\tablefirsthead
. . .
\StopDefiningTabulars
```

See section 8.10.1.

⚠ lateximage supertabular and xtab are not supported inside a `lateximage`.

```
\newcommand{\LWRST@firsthead}{}
```

```
3
4 \newcommand{\tablefirsthead}[1]{%
5     \long\gdef\LWRST@firsthead{#1}%
6 }
7
8 \newcommand{\tablehead}[1]{}
9 \newcommand{\tabletail}[1]{}
10
11 \newcommand{\LWRST@lasttail}{}
12
13 \newcommand{\tablelasttail}[1]{%
14     \long\gdef\LWRST@lasttail{#1}%
15 }


16 \newcommand{\tablecaption}[2][]{%
17     \long\gdef\LWRST@caption{%
18         \ifblank{#1}%
19             {\caption{#2}}%
20             {\caption[#1]{#2}}%
21     }%
22 }
23
24 \let\topcaption\tablecaption
25 \let\bottomcaption\tablecaption


26 \newcommand*{\LWRST@caption}{}
27
28 \newcommand*{\shrinkheight}[1]{}
29
30 \NewDocumentEnvironment{supertabular}{s o m}
31 {%
32 \LWR@traceinfo{supertabular}%
33 \begin{table}%
34 \LWRST@caption%
35 \begin{tabular}{#3}%
36 \TabularMacro\ifdefvoid{\LWRST@firsthead}%
37 {\LWR@getmynexttoken}%
38 {\expandafter\LWR@getmynexttoken\LWRST@firsthead}%
39 }%
40 {%
41 \ifdefvoid{\LWRST@lasttail}%
42 {}%
43 {%
44 \TabularMacro\ResumeTabular%
45 \LWRST@lasttail%
46 }%
47 \end{tabular}%
48 \end{table}%


49 \gdef\LWRST@caption{}%


50 \LWR@traceinfo{supertabular done}%
51 }
52
53 \NewDocumentEnvironment{mpsupertabular}{s o m}
54 {\minipage{\linewidth}\supertabular{#3}}
55 {\endsupertabular\endminipage}
```

File 479 **lwarp-svg.sty**

§ 588 Package svg

(*Emulates or patches code by* Philip Ilten, Falk Hanisch.)

svg (*Pkg*) svg is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{svg}[2020/10/23]

\xpretocmd{\includesvg}%
    {\begin{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includesvg}}

\xapptocmd{\includesvg}%
    {\end{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includesvg}}

\xpretocmd{\includeinkscape}%
    {\begin{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includeinkscape}}

\xapptocmd{\includeinkscape}%
    {\end{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includeinkscape}}
```

File 480 **lwarp-swfigure.sty**

§ 589 Package swfigure

(*Emulates or patches code by* Claudio Beccari.)

swfigure (*Pkg*) swfigure is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{swfigure}[2020-11-10]

\NewDocumentEnvironment{DFimage}%
  {O{SW} m O{#4} m o D(){0.8} D<>{0} D||{0.25} D!!{}}%
{%
    \begin{figure}
    \centering
    \includegraphics{#2}
    \caption[#3]{#4}
    \IfValueT{#5}{\label{#5}}
    \end{figure}
}%
{}%
```

File 481 **lwarp-sympytex.sty**

§ 590 Package sympytex

(*Emulates or patches code by* Tim Molteno.)

sympytex (*Pkg*) sympytex is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{sympytex}[2014/05/16]

\AfterEndPreamble{

\AtBeginEnvironment{sympyblock}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
            \LWR@forcenewpage%
            \LWR@atbeginverbatim{verbatim}%
        }%
}

\AfterEndEnvironment{sympyblock}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {%
            \LWR@afterendverbatim%
        }%
}

}
```

File 482 **lwarp-syntonly.sty**

§ 591 Package syntonly

(*Emulates or patches code by* Frank Mittelbach, Rainer Schöpf.)

syntonly (*Pkg*) syntonly is ignored.

for HTML output: Discard all options for lwarp-syntonly:

```
\LWR@ProvidesPackageDrop{syntonly}[2017/06/30]

\newif\ifsyntax@
\syntax@false

\newcommand*{\syntaxonly}{}

\@onlypreamble\syntaxonly


\def\nopages@{}
```

File 483 **lwarp-tabfigures.sty**

§ 592 Package tabfigures

tabfigures (*Pkg*) tabfigures is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tabfigures}[2012/01/24]
```

File 484 **lwarp-tablefootnote.sty**

§ 593 Package tablefootnote

tablefootnote (*Pkg*) tablefootnote is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tablefootnote}[2014/01/26]
```

This works because in HTML tables are no longer floats.

```
\LetLtxMacro\tablefootnote\footnote
```

File 485 **lwarp-tabls.sty**

§ 594 Package tabls

(*Emulates or patches code by* Donald Arseneau.)

tabls (*Pkg*) tabls is emulated. `\LWR@hline` is used to handle the optional argument when tabls is loaded.

for HTML output:

```
\LWR@ProvidesPackageDrop{tabls}
```

```
\newdimen\tablinesep
\newdimen\arraylinesep
\newdimen\extrarulesep
```

File 486 **lwarp-tabularx.sty**

§ 595 Package tabularx

(*Emulates or patches code by* David Carlisle.)

tabularx (*Pkg*) tabularx is emulated by lwarp.

for HTML output: Discard all options for lwarp-tabularx:

```
\LWR@ProvidesPackageDrop{tabularx}[2016/02/03]
\RequirePackage{array}
```

`\tabularxcolumn` is ignored. All `X` columns will be `p` for now. The width is ignored.

```
\def\tabularxcolumn#1{p{#1}}
\newcolumntype{X}{p{1in}}

\DeclareDocumentEnvironment{tabularx}{m o m}
    {\tabular{#3}}
    {\endtabular}

\DeclareDocumentEnvironment{tabularx*}{m o m}
    {\tabular{#3}}
    {\endtabular}
```

File 487 **lwarp-tabulary.sty**

§ 596 Package tabulary

(*Emulates or patches code by* David Carlisle.)

`tabulary` (*Pkg*) tabulary is emulated by lwarp.

for HTML output: Discard all options for lwarp-tabulary.

Column types `L`, `C`, `R`, and `J` are emulated by lwarp core code.

```
\LWR@ProvidesPackageDrop{tabulary}[2014/06/11]
\RequirePackage{array}

\NewDocumentEnvironment{tabulary}{m o m}
{\tabular{#3}}
{\endtabular}

\NewDocumentEnvironment{tabulary*}{m o m}
{\tabular{#3}}
{\endtabular}


\newcolumntype{L}{l}
\newcolumntype{C}{c}
\newcolumntype{R}{r}
\newcolumntype{J}{l}

\newdimen\tymin
\newdimen\tymax
\def\tyformat{}
```

File 488 **lwarp-tagpdf.sty**

§ 597 Package tagpdf

`tagpdf` (*Pkg*) tagpdf adds alt text, for images only. (HTML only has alternate text for images.)

The overall strategy is that tagpdf is deactivated, and slightly patched to process alt tags. Also see tagpdf-base, tagpdf-mc-code-generic, and tagpdf-mc-code-lua, following tagpdf.

for HTML output:

```
\LWR@ProvidesPackagePass{tagpdf}[2022-08-24]

\ExplSyntaxOn

\keys_define:nn { __tag / struct }
  {
    alt .code:n      = % Alt property
      {
%        \str_set_convert:Noon
%          \l__tag_tmpa_str
%          { #1 }
%          { default }
%          { utf16/hex }
%        \__tag_prop_gput:cnx
%          { g__tag_struct_\int_eval:n {\c@g__tag_struct_abs_int}_prop }
%          { Alt }
%          { <\l__tag_tmpa_str> }
        \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%      lwarp
      },
  }

\ExplSyntaxOff
```

The package is deactivated on load, and also each time `\tagpdfsetup` is used.

```
\LWR@tagpdf@deactivate
```

File 489 **lwarp-tagpdf-base.sty**

§ 598 Package tagpdf-base

(*Emulates or patches code by* Ulrike Fischer.)

`tagpdf-base` (*Pkg*) tagpdf-base is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{tagpdf-base}[2022-08-24]

\ExplSyntaxOn

\newcommand*{\LWR@tagpdf@deactivate}{
    \keys_set:nn { __tag / setup } {
        activate-space = false ,
        activate-mc = false ,
        activate-tree = false ,
        activate-struct = false
    }
}

\RenewDocumentCommand \tagpdfsetup { m }{
    \keys_set:nn { __tag / setup } { #1 }
    \LWR@tagpdf@deactivate
}
```

```
17
18 \RenewDocumentCommand \tagmcbegin { m }
19   {
20 %    \tag_mc_begin:n {#1}
21     \keys_set:nn { __tag / mc } {#1}
22   }
23
24 \RenewDocumentCommand \tagmcend {  }
25   {
26 %    \tag_mc_end:
27     \ThisAltText{}%     lwarp
28   }
29
30 \RenewDocumentCommand \tagmcuse { m }
31   {
32 %    \tag_mc_use:n {#1}
33   }
34
35 \RenewDocumentCommand \tagstructbegin { m }
36   {
37     \keys_set:nn { __tag / struct} { #1 }%      lwarp
38 %    \tag_struct_begin:n {#1}
39   }
40
41 \RenewDocumentCommand \tagstructend {  }
42   {
43 %   \tag_struct_end:
44   \ThisAltText{}%       lwarp
45   }
46
47 \RenewDocumentCommand \tagstructuse { m }
48   {
49 %    \tag_struct_use:n {#1}
50   }
51
52 \ExplSyntaxOff
```

File 490 **lwarp-tagpdf-mc-code-generic.sty**

§ 599 Package

tagpdf-mc-code-generic

(*Emulates or patches code by* Ulrike Fischer.)

tagpdf-mc-code-generic (*Pkg*) tagpdf-mc-code-generic is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{tagpdf-mc-code-generic}[2022-08-24]

2 \ExplSyntaxOn
3
4 % From tagpdf-mc-code-generic.sty:
5 \keys_define:nn { __tag / mc }
6   {
7     alt .code:n      = % Alt property
8       {
9 %        \str_set_convert:Noon
10 %          \l__tag_tmpa_str
11 %          { #1 }
```

```
%          { default }
%          { utf16/hex }
%        \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
%       \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
         \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%      lwarp
      },
  }

\ExplSyntaxOff
```

File 491 **lwarp-tagpdf-mc-code-lua.sty**

§ 600 Package tagpdf-mc-code-lua

(*Emulates or patches code by* Ulrike Fischer.)

tagpdf-mc-code-lua (*Pkg*) tagpdf-mc-code-lua is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{tagpdf-mc-code-lua}[2022-08-24]

\ExplSyntaxOn

\keys_define:nn { __tag / mc }
  {
    alt .code:n      = % Alt property
      {
%        \str_set_convert:Noon
%          \l__tag_tmpa_str
%          { #1 }
%          { default }
%          { utf16/hex }
%        \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
%       \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
%        \lua_now:e
%          {
%            ltx.__tag.func.store_mc_data
%              (
%           \__tag_get_mc_abs_cnt:,"alt","/Alt~<\str_use:N \l__tag_tmpa_str>"
%              )
%          }
         \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%      lwarp
      },
  }

\ExplSyntaxOff
```

File 492 **lwarp-tascmac.sty**

§ 601 Package tascmac

tascmac (*Pkg*) tascmac is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{tascmac}[2018/03/09]
```

```
\newenvironment*{boxnote}
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid black ;
            border-top: 1px dashed black ;
        ]{boxnote}
    }
    {\endBlockClass}

\newenvironment*{screen}[1][]
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid gray ;
            border-radius: 8pt
        ]{boxnote}
    }
    {\endBlockClass}

\newenvironment*{itembox}[2][]
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid gray ;
            border-radius: 8pt
        ]{boxnote}
        \InlineClass{itemboxtitle}{#2}\par
    }
    {\endBlockClass}

\newenvironment*{shadebox}
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid black ;
            box-shadow: 3px 3px 3px \#808080 ;
        ]{boxnote}
    }
    {\endBlockClass}

\newcommand*{\mask}[2]{%
    \InlineClass[background: lightgray]{mask}{#1}%
}

\newcommand*{\maskbox}[5]{%
    \InlineClass[background: lightgray]{mask}{#5}%
}

\newcommand*{\Maskbox}[6]{%
    \InlineClass[
        background: lightgray ;
        border: #5 solid black
    ]{mask}{#6}%
}

\newcommand*{\keytop}[2][]{%
    \InlineClass[%
        padding: .2ex ;
        border: 1px solid black ;
```

```
        border-radius: .7ex ;
    ]{keytop}{#2}%
}

\def\yen{\HTMLunicode{00A5}}

\def\return{\HTMLunicode{23CE}}

\def\Return{\HTMLunicode{23CE}}

\def\ascii{ASCII Corporation}

\def\Ascii{ASCII Corporation}

\def\ASCII{ASCII Corporation}
```

File 493 **lwarp-tcolorbox.sty**

§ 602 Package **tcolorbox**

(*Emulates or patches code by* Thomas F. Sturm.)

tcolorbox (*Pkg*) tcolorbox is patched for use by lwarp.

See section 8.3.8 for limitations.

for HTML output:

```
\LWR@ProvidesPackagePass{tcolorbox}[2023/02/12]

\newbool{LWR@havetcblower}
\boolfalse{LWR@havetcblower}
```

Colors are supported via HTML styles:

```
\newcommand{\LWR@tcolorbox@findcolors}{%
    \convertcolorspec{named}{tcbcolback}{HTML}\LWR@tcbcolback
    \convertcolorspec{named}{tcbcolframe}{HTML}\LWR@tcbcolframe
    \iftcb@titlefilled%
        \convertcolorspec{named}{tcbcolbacktitle}{HTML}\LWR@tcbcolbacktitle
    \else
        \convertcolorspec{named}{tcbcolframe}{HTML}\LWR@tcbcolbacktitle
    \fi
    \convertcolorspec{named}{tcbcoltitle}{HTML}\LWR@tcbcoltitle
    \convertcolorspec{named}{tcbcolupper}{HTML}\LWR@tcbcolupper
    \convertcolorspec{named}{tcbcollower}{HTML}\LWR@tcbcollower
}

\newcommand*{\LWR@tcolorbox@titlecolorstyles}{%
    border-top: 1px solid \LWR@origpound\LWR@tcbcolframe ;
    border-bottom: 1px solid \LWR@origpound\LWR@tcbcolframe ;
    background: \LWR@origpound\LWR@tcbcolbacktitle ;
    color: \LWR@origpound\LWR@tcbcoltitle ;
}
```

The title is placed inside its own of class tcolorboxtitle.

```
\newcommand*{\LWR@showtitle@}[1]{%
```

```
    \begin{BlockClass}[
        \LWR@tcolorbox@titlecolorstyles
    ]{tcolorboxtitle}
%                       \cmdKV@LWRtcolorbox@title\par
    \kvtcb@before@title#1\kvtcb@after@title
    \end{BlockClass}
}
```

If no title, a non-breakable space is used to take some vertical space.

```
\newcommand*{\LWR@showtitle}[1]{%
    \iftcb@titlevisible
    \LWR@showtitle@{#1}
    \else
    \LWR@showtitle@{~}
    \fi
}

\newcommand*{\LWR@tcolorbox@dophantom}{%
%     \sbox\tcb@phantombox{\kvtcb@phantom}%
%     \iftcb@hasPhantom%
%         \box\tcb@phantombox%
%         \tcb@hasPhantomfalse%
%     \fi%
    \kvtcb@phantom
    \let\kvtcb@phantom\@empty%
}
```

The tcolorbox is placed inside an external of class #1, which is tcolorbox or tcolorbox inlineminipage. The upper and lower parts are placed into their own internal s of class tcolorboxupper and tcolorboxlower.

```
\newcommand*{\LWR@tcolorboxstart}[1]{
    \LWR@tcolorbox@findcolors
    \begin{BlockClass}[
        border: 1px solid \LWR@origpound\LWR@tcbcolframe ;
        background: \LWR@origpound\LWR@tcbcolback ;
    ]{#1}
    \LWR@tcolorbox@dophantom%
    \ifdefvoid{\kvtcb@title}
        {}
        {
            \LWR@showtitle{\kvtcb@title}
        }
    \begin{BlockClass}[
        color: \LWR@origpound\LWR@tcbcolupper ;
    ]{tcolorboxupper}
}
```

Floats enclose the tcolorbox.

```
\newcommand*{\LWR@tcolorbox@dostartfloat}{%
    \ifx\kvtcb@float\@empty%
%         \tcb@set@normal@unbroken@beforeafter%
    \else%
%         \edef\tcb@before@unbroken{%
%             \noexpand\tcb@float@env@begin{tcbfloat}[\kvtcb@float]%
%             \noexpand\kvtcb@everyfloat%
%         }%
```

```
%         \let\tcb@after@unbroken=\tcb@float@env@end%
        \tcb@float@env@begin{tcbfloat}[\kvtcb@float]
        \noexpand\kvtcb@everyfloat
    \fi%
}

\newcommand*{\LWR@tcolorbox@doendfloat}{%
    \ifx\kvtcb@float\@empty%
    \else%
        \tcb@float@env@end%
    \fi%
}
```

Footnotes are handled via the main footnote mechanism, and pending notes are printed before and after each `tcolorbox`. Footnote numbering will not match the print output.

```
\renewenvironment{tcolorbox}[1][]
    {
        \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {
            \PackageError{lwarp}
                {%
             Lwarp cannot process a tcolorbox inside a lateximage\MessageBreak
                    or SVG math.\MessageBreak
                    Enter 'H' for possible solutions%
                }
                {%
                    Use \protect\tcbox, \protect\tcboxmath, or
                    \protect\tcbhighmath\space instead.\MessageBreak%
                    (Inside math, you probably want to use these anyhow.)%
                }
        }{\relax}
        \LWR@printpendingfootnotes
        \tcb@layer@inc
        \tcb@apply@box@options{#1}
        \LWR@tcolorbox@dostartfloat%
%         \tcbset{title=,#1}
        \boolfalse{LWR@havetcblower}
        \LWR@tcolorboxstart{tcolorbox}
        \tcb@insert@before@upper%
    }
    {
        \ifbool{LWR@havetcblower}{%
            \tcb@insert@after@lower%
        }{%
            \tcb@insert@after@upper%
        }%
        \end{BlockClass}
        \LWR@printpendingfootnotes
        \tcb@layer@dec
        \end{BlockClass}
        \LWR@tcolorbox@doendfloat%
    }
```

For the lower part, the upper part is finished then the lower is started. `\tcblower` is only temporarily defined where appropriate, so the HTML version is defined globally via `\newcommand` instead of `\renewcommand`.

```
\newcommand{\tcblower}{
    \tcb@insert@after@upper%
    \end{BlockClass}
    \begin{BlockClass}[%
        border-top: 1px dashed \LWR@origpound\LWR@tcbcolframe ;
        color: \LWR@origpound\LWR@tcbcollower ;
    ]{tcolorboxlower}
    \tcb@insert@before@lower%
}
```

Starred and unstarred `\tcbline` are simple `\hrule`s.

```
\AtBeginDocument{
\ifdef{\tcbline}{
    \newcommand*{\LWR@sub@tcbline}{%
        \begin{BlockClass}{hrule}
        \end{BlockClass}
    }
    \newcommand{\LWR@HTML@tcbline}{\@ifstar\LWR@sub@tcbline\LWR@sub@tcbline}
    \LWR@formatted{tcbline}
}{}
}

\newcommand{\LWR@HTML@tcbox}[2][]{
    \LWR@printpendingfootnotes
    \LWR@tcolorbox@dostartfloat%
    \begingroup
    \tcb@layer@inc
    \tcb@apply@box@options{#1}
%     \tcbset{title=,#1}
    \boolfalse{LWR@havetcblower}
    \LWR@tcolorboxstart{tcolorbox inlineminipage}
    \tcb@insert@before@upper%
    #2
    \ifbool{LWR@havetcblower}{%
        \tcb@insert@after@lower%
    }{%
        \tcb@insert@after@upper%
    }%
    \end{BlockClass}
    \LWR@printpendingfootnotes
    \end{BlockClass}
    \tcb@layer@dec%
    \endgroup%
    \LWR@tcolorbox@dostartfloat%
    \global\booltrue{LWR@minipagethispar}%
}
\LWR@formatted{tcbox}

\appto\LWR@restoreMathJaxformatting{%
    \renewcommand{\tcbox}[2][]{#2}%
}
```

Patches for the subtitle, which is placed inside a `<div>` of class `tcolorboxsubtitle`.

```
\xpatchcmd{\tcbsubtitle}
    {\begingroup}
    {\begingroup\let\kvtcb@title\relax\begin{BlockClass}{tcolorboxsubtitle}}
    {}
    {\LWR@patcherror{tcolorbox}{tcbsubtitle}}
```

```
\xpatchcmd{\tcbsubtitle}
    {\endgroup}
    {\end{BlockClass}\endgroup}
    {}
    {\LWR@patcherror{tcolorbox}{tcbsubtitleB}}
```

`\tcboxfit` is the same as `\tcbox`.

```
\AtBeginDocument{
    \ifdef{\tcboxfit}{%
        \let\LWR@HTML@tcboxfit\tcbox%
        \LWR@formatted{tcboxfit}
    }{}
}
```

`\tcbtitle` is patched to support the text font.

```
\LetLtxMacro\LWR@HTML@tcbtitle\tcbtitle
\xpatchcmd{\LWR@HTML@tcbtitle}
    {\tcb@insert@before@title\tcbtitletext}
  {\tcb@insert@before@title\LWR@textcurrentfont{\LWR@textcurrentcolor{\tcbtitletext}}}
    {}
    {\LWR@patcherror{tcolorbox}{LWR@HTML@tcbtitle}}
\LWR@formatted{tcbtitle}
```

List-of:

```
\renewcommand*\l@tcolorbox[2]{\hypertocfloat{1}{tcolorbox}{lof}{#1}{#2}}
```

Theorem limitations. An error is printed if the document uses `math`, `ams equation`, etc. `\tcboxmath` and `\tcbhighmath` are ignored for HTML.

```
\AtBeginDocument{
\pgfkeysifdefined{/tcb/libload/theorems}{

    \def\LWR@HTML@tcb@hack@amsmath{%
        \PackageError{lwarp}
            {%
              tcolorbox ``math'', ``ams equation'', and related\MessageBreak
                are not supported.\MessageBreak
                \protect\tcboxmath\space and
                \protect\tcbhighmath\space are emulated.\MessageBreak
                Enter 'H' for possible solutions%
            }
            {%
             Remove tcolorbox math-related options, and instead\MessageBreak
                use the usual math environments inside each tcolorbox.%
            }
    }
    \LWR@formatted{tcb@hack@amsmath}

    % Cause an error if using math:
    \tcbset{%
        math upper/.style={before upper*=\tcb@hack@amsmath,after upper*=$},%
        math lower/.style={before lower*=\tcb@hack@amsmath,after lower*=$},%
    }

    \appto\LWR@restoreorigformatting{%
```

```
    \tcbset{%
        math upper/.style={before upper*=$\displaystyle,after upper*=$},%
        math lower/.style={before lower*=$\displaystyle,after lower*=$},%
    }%
    }

    \newcommand{\LWR@HTML@tcboxmath}[2][]{#2}
    \LWR@formatted{tcboxmath}
    \newcommand{\LWR@HTML@tcbhighmath}[2][]{#2}
    \LWR@formatted{tcbhighmath}
    \appto\LWR@restoreMathJaxformatting{%
        \renewcommand{\tcboxmath}[2][]{#2}%
        \renewcommand{\tcbhighmath}[2][]{#2}%
    }
}{}% theorems loaded
}% AtBeginDocument
```

For MathJax:

```
\CustomizeMathJax{\newcommand{\tcbset}[1]{}}
\CustomizeMathJax{\newcommand{\tcbsetforeverylayer}[1]{}}
\CustomizeMathJax{\newcommand{\tcbox}[2][]{\boxed{\text{#2}}}}
\CustomizeMathJax{\newcommand{\tcboxfit}[2][]{\boxed{#2}}}
\CustomizeMathJax{\newcommand{\tcblower}{}}
\CustomizeMathJax{\newcommand{\tcbline}{}}
\CustomizeMathJax{\newcommand{\tcbtitle}{}}
\CustomizeMathJax{\newcommand{\tcbsubtitle[2][]{\mathrm{#2}}}}
\CustomizeMathJax{\newcommand{\tcboxmath}[2][]{\boxed{#2}}}
\CustomizeMathJax{\newcommand{\tcbhighmath}[2][]{\boxed{#2}}}
```

File 494 **lwarp-tensor.sty**

§ 603 Package

tensor

(*Emulates or patches code by* Philip G. Ratcliffe.)

tensor (*Pkg*) tensor is used as-is for SVG math, and is emulated for MathJax.

⚠ spacing Compressed spacing and left justification are not possible with MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{tensor}[2004/12/20]
```

For MathJax. Special handling is required to parse the superscript and subscript arguments.

When a superscript or subscript is seen, it is processed and then the remainder is processesed recursively.

```
\begin{warpMathJax}
\CustomizeMathJax{\def\LWRtensorindicesthreesub#1#2{{_{#2}}\LWRtensorindicesthree}}
\CustomizeMathJax{\def\LWRtensorindicesthreesup#1#2{{^{#2}}\LWRtensorindicesthree}}
```

If not a superscript nor a subscript, processing stops.

```
\CustomizeMathJax{\newcommand{\LWRtensorindicesthreenotsup}{}}
```

Check ahead for a superscript or a subscript.

```
\CustomizeMathJax{\newcommand{\LWRtensorindicesthreenotsub}{
    \ifnextchar ^ \LWRtensorindicesthreesup \LWRtensorindicesthreenotsup
}}

\CustomizeMathJax{\newcommand{\LWRtensorindicesthree}{
    \ifnextchar _ \LWRtensorindicesthreesub \LWRtensorindicesthreenotsub
}}
```

Ignore star.

```
\CustomizeMathJax{\newcommand{\LWRtensorindicestwo}{
    \ifstar\LWRtensorindicesthree\LWRtensorindicesthree
}}
```

Remove the outer brace of the argument.

```
\CustomizeMathJax{\newcommand{\indices}[1]{\LWRtensorindicestwo#1}}
```

Attempting to use \vphantom here does not work:

```
\CustomizeMathJax{\newcommand{\LWRtensortwo}[3][]{{}\indices{#1}{#2}\indices{#3}}}
```

Ignore star.

```
\CustomizeMathJax{\newcommand{\tensor}{\ifstar\LWRtensortwo\LWRtensortwo}}
```

In text mode, \nuclide is converted to an SVG image.

```
\CustomizeMathJax{%
    \newcommand{\LWRnuclidetwo}[2][]{%
        {%
            \vphantom{\mathrm{#2}}%
            {}^{\LWRtensornucleonnumber}_{#1}%
            \mathrm{#2}%
        }%
    }%
}

\CustomizeMathJax{%
    \newcommand{\nuclide}[1][]{%
        \def\LWRtensornucleonnumber{#1}%
        \LWRnuclidetwo%
    }%
}
\end{warpMathJax}
```

File 495 **lwarp-termcal.sty**

§ 604 Package termcal

(*Emulates or patches code by* Bill Mitchell.)

`termcal` (*Pkg*) termcal is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{termcal}% questionable date in the .sty file
```

Nullify the @ because everything is being done in a token list.

```
\xpatchcmd{\endcalendar}
    {@{}}
    {}
    {}
    {\LWR@patcherror{termcal}{endcalendar}}
```

Remove the hbox:

```
\xpatchcmd{\ca@doaday}
    {\hbox to \hsize{\calprintdate\hfill\ifclassday\calprintclass\fi}}
    {%
        \calprintdate\hfill\ifclassday\calprintclass\fi%
    }
    {}
    {\LWR@patcherror{termcal}{ca@doaday}}
```

Change each of two ampersands to call the lwarp tabular version:

```
\xpatchcmd{\calday}
    {&}
    {\LWR@tabularampersand}
    {}
    {\LWR@patcherror{termcal}{calday}}

\xpatchcmd{\calday}
    {&}
    {\LWR@tabularampersand}
    {}
    {\LWR@patcherror{termcal}{calday B}}
```

File 496 **lwarp-textarea.sty**

§ 605 Package textarea

(*Emulates or patches code by* Alexander I. Rozhenko.)

`textarea` (*Pkg*) textarea is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{textarea}[2005/12/26]

\newcommand\StartFromTextArea{}
\newcommand\StartFromHeaderArea{}
\newcommand*\RestoreTextArea{}
\newcommand*\ExpandTextArea[1][*]{}
\let\NCC@restoretextarea\@empty
```

File 497 **lwarp-textcomp.sty**

§ 606 Package textcomp

(*Emulates or patches code by* Frank Mittelbach, Robin Fairbairns, Werner Lemberg.)

textcomp (*Pkg*) textcomp is patched for use by lwarp.

For MathJax, the MathJax packge is used.

§ 606.1 Limitations

Some textcomp symbols do not have Unicode equivalents, and thus are not supported.

⚠ missing symbols Many textcomp symbols are not supported by many system / browser fonts. In the CSS try referencing fonts which are more complete, but expect to see gaps in coverage.

§ 606.2 Package loading

for HTML output:

```
\LWR@ProvidesPackagePass{textcomp}[2017/04/05]
```

§ 606.3 HTML symbols

For HTML, use HTML entities or direct Unicode, depending on the engine.

`\AtBeginDocument` improves support for LuaLaTeX and XeLaTeX.

§ 606.3.1 pdfLaTeX symbols

```
\AtBeginDocument{
\ifPDFTeX% pdflatex or dvi latex
\newcommand*{\LWR@HTML@textdegree}{\HTMLentity{deg}}
\newcommand*{\LWR@HTML@textcelsius}{\HTMLunicode{2103}}
\newcommand*{\LWR@HTML@textohm}{\HTMLunicode{2126}}
\newcommand*{\LWR@HTML@textmu}{\HTMLunicode{00B5}}
\newcommand*{\LWR@HTML@textlquill}{\HTMLunicode{2045}}
\newcommand*{\LWR@HTML@textrquill}{\HTMLunicode{2046}}
\newcommand*{\LWR@HTML@textcircledP}{\HTMLunicode{2117}}
\newcommand*{\LWR@HTML@texttwelveudash}{\HTMLunicode{2014}}% emdash
\newcommand*{\LWR@HTML@textthreequartersemdash}{\HTMLunicode{2014}}% emdash
\newcommand*{\LWR@HTML@textmho}{\HTMLunicode{2127}}
\newcommand*{\LWR@HTML@textnaira}{\HTMLunicode{20A6}}
\newcommand*{\LWR@HTML@textpeso}{\HTMLunicode{20B1}}
\newcommand*{\LWR@HTML@textrecipe}{\HTMLunicode{211E}}
\newcommand*{\LWR@HTML@textinterrobang}{\HTMLunicode{203D}}
\newcommand*{\LWR@HTML@textinterrobangdown}{\HTMLunicode{2E18}}
\newcommand*{\LWR@HTML@textperthousand}{\HTMLunicode{2030}}
\newcommand*{\LWR@HTML@textpertenthousand}{\HTMLunicode{2031}}
\newcommand*{\LWR@HTML@textbaht}{\HTMLunicode{0E3F}}
\newcommand*{\LWR@HTML@textdiscount}{\%}
\newcommand*{\LWR@HTML@textservicemark}{\HTMLunicode{2120}}
\else
```

§ 606.3.2 XeLaTeX and LuaLaTeX symbols

NOTE: Some of the following do not print well in the listing. Consult the .dtx or .sty file for the actual characters.

```
\newcommand*{\LWR@HTML@textdegree}{°}
\newcommand*{\LWR@HTML@textcelsius}{℃}
\newcommand*{\LWR@HTML@textohm}{Ω}
```

```
\newcommand*{\LWR@HTML@textmu}{µ}
\newcommand*{\LWR@HTML@textlquill}{{}
\newcommand*{\LWR@HTML@textrquill}{}}
\newcommand*{\LWR@HTML@textcircledP}{Ⓟ}
\newcommand*{\LWR@HTML@texttwelveudash}{–}% emdash
\newcommand*{\LWR@HTML@textthreequartersemdash}{–}% emdash
\newcommand*{\LWR@HTML@textmho}{℧}
\newcommand*{\LWR@HTML@textnaira}{₦}
\newcommand*{\LWR@HTML@textpeso}{₱}
\newcommand*{\LWR@HTML@textrecipe}{℞}
\newcommand*{\LWR@HTML@textinterrobang}{‽}
\newcommand*{\LWR@HTML@textinterrobangdown}{⸘}
\newcommand*{\LWR@HTML@textperthousand}{‰}
\newcommand*{\LWR@HTML@textpertenthousand}{‱}
\newcommand*{\LWR@HTML@textbaht}{฿}
\newcommand*{\LWR@HTML@textdiscount}{\%}
\newcommand*{\LWR@HTML@textservicemark}{℠}
\fi

\LWR@formatted{textdegree}
\LWR@formatted{textcelsius}
\LWR@formatted{textohm}
\LWR@formatted{textmu}
\LWR@formatted{textlquill}
\LWR@formatted{textrquill}
\LWR@formatted{textcircledP}
\LWR@formatted{texttwelveudash}
\LWR@formatted{textthreequartersemdash}
\LWR@formatted{textmho}
\LWR@formatted{textnaira}
\LWR@formatted{textpeso}
\LWR@formatted{textrecipe}
\LWR@formatted{textinterrobang}
\LWR@formatted{textinterrobangdown}
\LWR@formatted{textperthousand}
\LWR@formatted{textpertenthousand}
\LWR@formatted{textbaht}
\LWR@formatted{textdiscount}
\LWR@formatted{textservicemark}
```

§ 606.4 HTML diacritics

For HTML, Unicode diacritical marks are used:

```
\newcommand*{\LWR@HTML@capitalcedilla}[1]{#1\HTMLunicode{0327}}
\newcommand*{\LWR@HTML@capitalogonek}[1]{#1\HTMLunicode{0328}}
\newcommand*{\LWR@HTML@capitalgrave}[1]{#1\HTMLunicode{0300}}
\newcommand*{\LWR@HTML@capitalacute}[1]{#1\HTMLunicode{0301}}
\newcommand*{\LWR@HTML@capitalcircumflex}[1]{#1\HTMLunicode{0302}}
\newcommand*{\LWR@HTML@capitaltilde}[1]{#1\HTMLunicode{0303}}
\newcommand*{\LWR@HTML@capitaldieresis}[1]{#1\HTMLunicode{0308}}
\newcommand*{\LWR@HTML@capitalhungarumlaut}[1]{#1\HTMLunicode{30B}}
\newcommand*{\LWR@HTML@capitalring}[1]{#1\HTMLunicode{30A}}
\newcommand*{\LWR@HTML@capitalcaron}[1]{#1\HTMLunicode{30C}}
\newcommand*{\LWR@HTML@capitalbreve}[1]{#1\HTMLunicode{306}}
\newcommand*{\LWR@HTML@capitalmacron}[1]{#1\HTMLunicode{304}}
\newcommand*{\LWR@HTML@capitaldotaccent}[1]{#1\HTMLunicode{307}}
```

\textcircled becomes a span with a rounded border. \providecommand is used to avoid conflict with xunicode.

```
\providecommand*{\LWR@HTML@textcircled}[1]{%
    \InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
}

\LWR@formatted{capitalcedilla}
\LWR@formatted{capitalogonek}
\LWR@formatted{capitalgrave}
\LWR@formatted{capitalacute}
\LWR@formatted{capitalcircumflex}
\LWR@formatted{capitaltilde}
\LWR@formatted{capitaldieresis}
\LWR@formatted{capitalhungarumlaut}
\LWR@formatted{capitalring}
\LWR@formatted{capitalcaron}
\LWR@formatted{capitalbreve}
\LWR@formatted{capitalmacron}
\LWR@formatted{capitaldotaccent}

\LWR@formatted{textcircled}
```

Nullify textcomp macros when generating filenames:

```
\FilenameNullify{%
    \renewcommand*{\textdegree}{}%
    \renewcommand*{\textcelsius}{}%
    \renewcommand*{\textohm}{}%
    \renewcommand*{\textmu}{}%
    \renewcommand*{\textlquill}{}%
    \renewcommand*{\textrquill}{}%
    \renewcommand*{\textcircledP}{}%
    \renewcommand*{\texttwelveudash}{}%
    \renewcommand*{\textthreequartersemdash}{}%
    \renewcommand*{\textmho}{}%
    \renewcommand*{\textnaira}{}%
    \renewcommand*{\textpeso}{}%
    \renewcommand*{\textrecipe}{}%
    \renewcommand*{\textinterrobang}{}%
    \renewcommand*{\textinterrobangdown}{}%
    \renewcommand*{\textperthousand}{}%
    \renewcommand*{\textpertenthousand}{}%
    \renewcommand*{\textbaht}{}%
    \renewcommand*{\textdiscount}{}%
    \renewcommand*{\textservicemark}{}%
    \renewcommand*{\textcircled}[1]{#1}%
    \renewcommand*{\capitalcedilla}[1]{#1}%
    \renewcommand*{\capitalogonek}[1]{#1}%
    \renewcommand*{\capitalgrave}[1]{#1}%
    \renewcommand*{\capitalacute}[1]{#1}%
    \renewcommand*{\capitalcircumflex}[1]{#1}%
    \renewcommand*{\capitaltilde}[1]{#1}%
    \renewcommand*{\capitaldieresis}[1]{#1}%
    \renewcommand*{\capitalhungarumlaut}[1]{#1}%
    \renewcommand*{\capitalring}[1]{#1}%
    \renewcommand*{\capitalcaron}[1]{#1}%
    \renewcommand*{\capitalbreve}[1]{#1}%
    \renewcommand*{\capitalmacron}[1]{#1}%
    \renewcommand*{\capitaldotaccent}[1]{#1}%
```

```
}% FilenameNullify

}% AtBeginDocument
```

For MathJax:

```
\CustomizeMathJax{\require{textcomp}}
```

File 498 **lwarp-textfit.sty**

§ 607 Package textfit

`textfit` (*Pkg*) textfit is emulated.

Text is placed into a of class `textfit`. Sizes are approximated, and also limited by browser min/max font-size settings.

for HTML output:

```
\LWR@ProvidesPackageDrop{textfit}[1994/04/15]

\newsavebox{\LWR@textfitbox}

\newcommand*{\LWR@textfitscale}[2]{%
\setlength{\LWR@templengthone}{#1}%
\setlength{\LWR@templengthone}{%
    1em*\ratio{\LWR@templengthone}{\LWR@templengthtwo}%
}%
\InlineClass[font-size:\LWR@printlength{\LWR@templengthone}]{textfit}{#2}%
}

\newcommand*{\scaletowidth}[2]{%
\sbox{\LWR@textfitbox}{#2}%
\settowidth{\LWR@templengthtwo}{\usebox{\LWR@textfitbox}}%
\LWR@textfitscale{#1}{#2}%
}

\newcommand*{\scaletoheight}[2]{%
\sbox{\LWR@textfitbox}{#2}%
\settoheight{\LWR@templengthtwo}{\usebox{\LWR@textfitbox}}%
\LWR@textfitscale{#1}{#2}%
}
```

File 499 **lwarp-textpos.sty**

§ 608 Package textpos

(*Emulates or patches code by* Norman Gray.)

`textpos` (*Pkg*) textpos is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{textpos}[2020/09/26]

\NewDocumentEnvironment{textblock}{m o r()}{}{}
\NewDocumentEnvironment{textblock*}{m o r()}{}{}
\newcommand*{\TPGrid}[3][]{}
```

```
\def\TPShowGrid{\@ifstar{\@TPShowGrid}{\@TPShowGrid}}
\def\@TPShowGrid#1#2{}
\NewDocumentCommand{\TPMargin}{s o}{}
\newcommand*{\textblockcolour}[1]{}
\newcommand*{\textblockrulecolour}[1]{}
\newcommand*{\textblockcolor}[1]{}
\newcommand*{\textblockrulecolor}[1]{}
\newcommand*{\tekstblokkulur}[1]{}
\newcommand*{\tekstblokrulekulur}[1]{}
\newlength{\TPHorizModule}
\newlength{\TPVertModule}
\newlength{\TPboxrulesize}
\newcommand{\textblocklabel}[1]{}
\newcommand*{\showtextsize}{}
\newcommand{\textblockorigin}[2]{}
\newcommand*{\TPoptions}[1]{}
\newcommand*{\TPReferencePosition}[1]{}
```

File 500 **lwarp-theorem.sty**

§ 609 Package theorem

(*Emulates or patches code by* Frank Mittelbach.)

theorem (*Pkg*) theorem is patched for use by lwarp.

Table 21: Theorem package — CSS styling of theorems and proofs

Theorem: of class theorembody<theoremstyle>

Theorem Header: of class theoremheader

where <theoremstyle> is plain, break, etc.

for HTML output:

```
\LWR@ProvidesPackagePass{theorem}[2014/10/28]
```

§ 609.1 Remembering the theorem style

Storage for the style being used for new theorems:

```
\newcommand{\LWR@newtheoremstyle}{plain}
```

Patched to remember the style being used for new theorems:

```
\gdef\theoremstyle#1{%
   \@ifundefined{th@#1}{\@warning
          {Unknown theoremstyle `#1'. Using `plain'}%
          \theorem@style{plain}%
           \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
          }%
      {%
          \theorem@style{#1}%
          \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
      }%
      \begingroup
```

```
        \csname th@\the\theorem@style \endcsname
      \endgroup}
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```
\gdef\@xnthm#1#2[#3]{%
   \expandafter\@ifdefinable\csname #1\endcsname
   {%
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \@definecounter{#1}\@newctr{#1}[#3]%
    \expandafter\xdef\csname the#1\endcsname
      {\expandafter \noexpand \csname the#3\endcsname
       \@thmcountersep \@thmcounter{#1}}%
    \def\@tempa{\global\@namedef{#1}}%
    \expandafter \@tempa \expandafter{%
      \csname th@\the \theorem@style
            \expandafter \endcsname \the \theorem@bodyfont
     \@thm{#1}{#2}}%
    \global \expandafter \let \csname end#1\endcsname \@endtheorem
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
   }}

\gdef\@ynthm#1#2{%
    \expandafter\@ifdefinable\csname #1\endcsname
   {
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \@definecounter{#1}%
    \expandafter\xdef\csname the#1\endcsname{\@thmcounter{#1}}%
    \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
     \expandafter{\csname th@\the \theorem@style \expandafter
     \endcsname \the\theorem@bodyfont \@thm{#1}{#2}}%
    \global \expandafter \let \csname end#1\endcsname \@endtheorem
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
   }}

\gdef\@othm#1[#2]#3{%
  \expandafter\ifx\csname c@#2\endcsname\relax
   \@nocounterr{#2}%
  \else
   \expandafter\@ifdefinable\csname #1\endcsname
   {
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \expandafter \xdef \csname the#1\endcsname
     {\expandafter \noexpand \csname the#2\endcsname}%
    \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
     \expandafter{\csname th@\the \theorem@style \expandafter
     \endcsname \the\theorem@bodyfont \@thm{#2}{#3}}%
    \global \expandafter \let \csname end#1\endcsname \@endtheorem
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
    }%
  \fi}
```

§ 609.2 css patches

The following are patched for css.

These were in individual files `thp.sty` for plain, `thmb.sty` for margin break, etc. They are gathered together here.

Each theorem is encased in a `BlockClass` environment of class `theorembody<style>`.

Each header is encased in an `\InlineClass` of class `theoremheader`.

```
\gdef\th@plain{%
  \def\@begintheorem##1##2{%
        \item[
            \InlineClass{theoremheader}{##1\ ##2}
        ]
    }%
\def\@opargbegintheorem##1##2##3{%
   \item[
        \InlineClass{theoremheader}{##1\ ##2\ (##3)}
    ]
    }
}

\gdef\th@break{%
  \def\@begintheorem##1##2{%
    \item[
        \InlineClass{theoremheader}{##1\ ##2}\newline%
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
  \item[
        \InlineClass{theoremheader}{##1\ ##2\ (##3)}\newline
    ]
    }
}

\gdef\th@marginbreak{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2 \qquad ##1}\newline
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
        \InlineClass{theoremheader}{##2 \qquad ##1\ %
        (##3)}\newline
    ]
    }
}

\gdef\th@changebreak{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2\ ##1}\newline
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
        \InlineClass{theoremheader}{ ##2\ ##1\ %
```

```
        (##3)}\newline
    ]
    }
}

\gdef\th@change{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2\ ##1}
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
        \InlineClass{theoremheader}{##2\ ##1\ (##3)}
    ]
    }
}

\gdef\th@margin{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2 \qquad ##1}
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
          \InlineClass{theoremheader}{##2 \qquad ##1\ (##3)}
    ]
    }
}
```

Patched for css:

```
\gdef\@thm#1#2{\refstepcounter{#1}%
\LWR@forcenewpage% lwarp

    \LWR@printpendingfootnotes%                   lwarp

    \BlockClass{theorembody\LWR@thisthmstyle}% lwarp
   \trivlist
   \@topsep \theorempreskipamount               % used by first \item
   \@topsepadd \theorempostskipamount           % used by \@endparenv
   \@ifnextchar [%
   {\@ythm{#1}{#2}}%
   {\@begintheorem{#2}{\csname the#1\endcsname}\ignorespaces}}

\gdef\@endtheorem{%
\endtrivlist

    \LWR@printpendingfootnotes%                   lwarp

\endBlockClass
}
```

File 501 **lwarp-thinsp.sty**

§ 610 Package thinsp

thinsp (*Pkg*) thinsp is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{thinsp}[2016/10/02]

\AtBeginDocument{
\let\thinthinspace\relax% defined by some packages
\newcommand*{\thinthinspace}{\thinspace}
}

\newcommand*{\stretchthinspace}{\thinspace}
\newcommand*{\stretchthinthinspace}{\thinthinspace}
\newcommand*{\stretchnegthinspace}{\negthinspace}
```

File 502 **lwarp-thm-listof.sty**

§ 611 Package thm-listof

(*Emulates or patches code by* Ulrich M. Schwarz, Yukai Chou.)

thm-listof (*Pkg*) thm-listof is part of thmtools, and is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{thm-listof}[2019/12/22]
```

For font control, see the generated HTML and use CSS per amsthm or ntheorem.

Other thm-* package may be loaded by thm-listof.

```
\IfPackageAtLeastTF{thm-listof}{2020/08/01}{% v0.72
  \def\thmtlo@newentry{%
    \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
  }
}{% earlier than v0.72
    \xpatchcmd{\listoftheorems}
        {%
            \@xa\protected@edef\csname l@\thmt@envname\endcsname{%
                \@nx\@dottedtocline{1}{1.5em}{\@nx\thmt@listnumwidth}%
            }%
        }
        {%
        \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
        }
        {}
        {\LWR@patcherror{thm-listof}{listoftheorems}}

    \xpatchcmd{\thmt@mklistcmd}
        {%
            \@xa\protected@edef\csname l@\thmt@envname\endcsname{%
                \@nx\@dottedtocline{1}{1.5em}{\@nx\thmt@listnumwidth}%
```

```
            }%
        }
        {%
        \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
        }
        {}
        {\LWR@patcherror{thm-listof}{thmt@mklistcmd}}
}
```

File 503 **lwarp-thm-restate.sty**

§ 612 Package thm-restate

(*Emulates or patches code by* Ulrich M. Schwarz.)

thm-restate (*Pkg*) thm-restate is part of thmtools, and is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{thm-restate}[2020/08/01]

\xpatchcmd{\thmt@restatable}
    {\@ifstar}
    {\edef\LWR@thisthmstyle{#2}\@ifstar}
    {}
    {\LWR@patcherror{thm-restate}{thmt@restatable}}
```

File 504 **lwarp-thmbox.sty**

§ 613 Package thmbox

(*Emulates or patches code by* Emmanuel Beffara.)

thmbox (*Pkg*) thmbox is emulated for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{thmbox}[2005/04/24]

\renewenvironment{thmbox}[2][]%
    {%
        \begin{BlockClass}{thmbox}
        \begin{BlockClass}{thmboxtitle}
        #2
        \end{BlockClass}
    }
    {\end{BlockClass}}

\renewenvironment{proof}[1][]
    {%
        \begin{BlockClass}{thmboxproof}%
        \InlineClass{thmboxproofname}{\proofname\ #1\unskip\,:}
    }
    {%
        \qquad\HTMLunicode{220E}
        \end{BlockClass}
    }

```

```
\renewenvironment{example}[1][\examplename]%
    {%
        \begin{BlockClass}{thmboxexample}%
        \InlineClass{thmboxexamplename}{#1\,:}
    }
    {\end{BlockClass}}

\renewenvironment{leftbar}[1][]%
    {\begin{BlockClass}{thmboxleftbar}}
    {\end{BlockClass}}
```

File 505 **lwarp-thmtools.sty**

§ 614 Package thmtools

(*Emulates or patches code by* Ulrich M. Schwarz.)

`thmtools` (*Pkg*) thmtools is patched for use by lwarp.

Also see thm-listof and thm-restate.

for HTML output:

```
\LWR@ProvidesPackagePass{thmtools}[2020/08/01]
```

The following patches either thm-amsthm or thm-ntheorem.

```
\def\thmt@headstyle@margin{%
    \InlineClass{amsthmnnumbertheorem}{\NUMBER}
    \
    \InlineClass{amsthmnametheorem}{\NAME}
    \InlineClass{amsthmnotetheorem}{\NOTE}
}

\let\thmt@headstyle@swapnumber\thmt@headstyle@margin
```

File 506 **lwarp-threadcol.sty**

§ 615 Package threadcol

`threadcol` (*Pkg*) threadcol is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{threadcol}[2013/01/06]

\newcommand{\setthreadname}[1]{}
```

File 507 **lwarp-threeparttable.sty**

§ 616 Package threeparttable

(*Emulates or patches code by* Donald Arseneau.)

`threeparttable` (*Pkg*) threeparttable is emulated.

Table note are contained inside a css of class tnotes. If enumitem is used, the note item labels are also individually highlighted with an additional css of class tnoteitemheader, otherwise they are plain text.

for HTML output:

```
\LWR@ProvidesPackageDrop{threeparttable}[2003/06/13]
```

Env threeparttable [⟨*alignment*⟩]

```
\newenvironment*{threeparttable}[1][b]
    {\def\@captype{table}}
    {}
```

Env tablenotes [⟨*options*⟩]

```
\newenvironment*{tablenotes}[1][]
{%
\LWR@forcenewpage
\BlockClass{tnotes}%

\description%
}
{%
\enddescription%
\endBlockClass%
}
```

\tnote {⟨*text*⟩}

```
\newcommand{\tnote}[1]{\LWR@htmlspan{sup}{#1}}
```

Env measuredfigure [⟨*alignment*⟩]

```
\newenvironment*{measuredfigure}[1][t]
    {\def\@captype{figure}}
    {}
```

File 508 **lwarp-threeparttablex.sty**

§ 617 Package

threeparttablex

threeparttablex (*Pkg*) threeparttablex is patched for use by lwarp.

threeparttablex is used with longtable and booktabs as follows:

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  \bottomrule \insertTableNotes \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{   % HTML last footer
  \bottomrule
  \UseMinipageWidths     % optional
  \insertTableNotes
  \endlastfoot
}
\end{longtable}
```

table width

The table notes are created using a \multicolumn. By default the width is not specified to the browser, so long table notes can cause the table to be spread out horizontally. For HTML output, lwarp guesses the width of the table depending on the number of columns, then restricts its guess to a min/max range. To use this guess for the width of the table notes, use \UseMinipageWidths before \insertTableNotes. The width is then specified, and in many cases the result is an improvement in overall table layout.

for HTML output:

```
\LWR@ProvidesPackagePass{threeparttablex}[2013/07/23]
```

The width is guessed depending on the number of columns, then limited to a min/max.

```
\renewcommand\insertTableNotes{%
    \setlength{\LWR@templengthone}{.375in*\value{LWR@tabletotalLaTeXcols}}%
    \setlength{\LWR@templengthone}{\minof{\textwidth}{\LWR@templengthone}}%
    \setlength{\LWR@templengthone}{\maxof{2.5in}{\LWR@templengthone}}%
  \multicolumn{\value{LWR@tabletotalLaTeXcols}}{c}{%
      \parbox{\LWR@templengthone}{%
        \begin{tablenotes}[\TPTL@optarg]%
          \TPTL@font%
          \TPTL@body%
        \end{tablenotes}%
      }%
  }%
}

\providecommand{\TPTL@tnotex}{}
\renewcommand{\TPTL@tnotex}[2]{\tnote{\nameref{#2}}}
```

File 509 **lwarp-thumb.sty**

§ 618 Package thumb

thumb (*Pkg*) thumb is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{thumb}[1997/12/24]

\newcommand*{\Overviewpage}{}
```

```
\newlength{\thumbheight}
\newlength{\thumbwidth}
```

File 510 **lwarp-thumbs.sty**

§ 619 Package

thumbs

thumbs (*Pkg*) thumbs is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{thumbs}[2014/03/09]
```

```
\newcommand{\addthumb}[4]{}
\newcommand{\addtitlethumb}[5]{}
\newcommand{\stopthumb}{}
\newcommand{\continuethumb}{}
\newcommand{\thumbsoverview}[1]{}
\newcommand{\thumbsoverviewback}[1]{}
\newcommand{\thumbsoverviewverso}[1]{}
\newcommand{\thumbsoverviewdouble}[1]{}
\newcommand{\thumbnewcolumn}{}
\newcommand{\addthumbsoverviewtocontents}[2]{}
\newcommand{\thumbsnophantom}{}
```

File 511 **lwarp-tikz.sty**

§ 620 Package

tikz

(*Emulates or patches code by* Till Tantau.)

tikz (*Pkg*) tikz is supported.

⚠ displaymath and matrices

If using display math with `tikzpicture` or `\tikz`, along with matrices with the `&` character, the document must be modified as follows:

```
\usepackage{tikz}
\tikzset{every picture/.style={ampersand replacement=\&}}
```

and each instance of `&` in the tikz expression must be replaced with `\&`.

Accept all options for lwarp-tikz:

```
\LWR@ProvidesPackagePass{tikz}[2015/08/07]
```

catcodes

lwarp changes the catcode of `$` for its own use. The Ti*k*Z `babel` library temporarily changes catcodes back to normal for Ti*k*Z's use. tikz v3.0.0 introduced the `babel` library which handles catcode changes. For older versions, lwarp must change `$`'s catcode itself.

Also see:
https://tex.stackexchange.com/questions/16199/test-if-a-package-or-package-option-is-loaded

```
\newbool{LWR@tikzbabel}
```

```
3
4 \IfPackageAtLeastTF{tikz}{2013/12/20}% Test for Tikz version v3.0.0
5 {\usetikzlibrary{babel}\booltrue{LWR@tikzbabel}}
6 {\boolfalse{LWR@tikzbabel}}
```

Env pgfpicture

The \pgfpicture environment is enclosed inside a \lateximage. Enclose the low-level \pgfpicture in a lateximage. This is also used by the higher-level \tikz and tikzpicture.

```
7 \preto\pgfpicture{%
8     \begin{lateximage}[-tikz-~\PackageDiagramAltText]%
9     \ifbool{LWR@tikzbabel}% Test for Tikz version v3.0.0
10    {}%
11    {\catcode`\$=3}% dollar sign is math shift
12 }
13
14 \appto\endpgfpicture{\end{lateximage}}
```

Ti*k*Z is placed inside an SVG image, so use the original meanings of the following:

```
15 \LetLtxMacro\pgfutil@minipage\LWR@print@minipage
16 \let\pgfutil@endminipage\endLWR@print@minipage
17
18 \let\pgfutil@raggedleft\LWR@print@raggedleft
19 \let\pgfutil@raggedright\LWR@print@raggedright


20 \def\pgfutil@font@tiny{\LWR@print@tiny}
21 \def\pgfutil@font@scriptsize{\LWR@print@scriptsize}
22 \def\pgfutil@font@footnotesize{\LWR@print@footnotesize}
23 \def\pgfutil@font@small{\LWR@print@small}
24 \def\pgfutil@font@normalsize{\LWR@print@normalsize}
25 \def\pgfutil@font@large{\LWR@print@large}
26 \def\pgfutil@font@Large{\LWR@print@Large}
27 \def\pgfutil@font@huge{\LWR@print@huge}
28 \def\pgfutil@font@Huge{\LWR@print@Huge}
29
30 \def\pgfutil@font@itshape{\LWR@print@itshape}
31 \def\pgfutil@font@bfseries{\LWR@print@bfseries}
32
33 \def\pgfutil@font@normalfont{\LWR@print@normalfont}
```

File 512 **lwarp-tikz-imagelabels.sty**

§ 621 Package

tikz-imagelabels

(*Emulates or patches code by* Tobias Plüss.)

tikz-imagelabels (*Pkg*) tikz-imagelabels is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{tikz-imagelabels}[2019/06/27]

2 \BeforeBeginEnvironment{annotationimage}{%
3     \begin{lateximage}[-tikz-imagelabels-~\PackageDiagramAltText]%
4 }
5
6 \AfterEndEnvironment{annotationimage}{\end{lateximage}}
```

File 513 **lwarp-titleps.sty**

§ 622 Package

titleps

(*Emulates or patches code by* Javier Bezos.)

titleps (*Pkg*) titleps is loaded and used by lwarp during HTML output. All user options and macros are ignored and disabled.

Discard all options for lwarp-titleps:

for HTML output:

```
\LWR@ProvidesPackageDrop{titleps}[2016/03/15]
```

\pagestyle and \thispagestyle are already disabled in the lwarp code.

\newpagestyle {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

```
\NewDocumentCommand{\newpagestyle}{m o m}{}
```

\renewpagestyle {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

```
\NewDocumentCommand{\renewpagestyle}{m o m}{}
```

\sethead [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

```
\NewDocumentCommand{\sethead}{o o o m m m}{}
```

\setfoot [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

```
\NewDocumentCommand{\setfoot}{o o o m m m}{}
```

\settitlemarks * {⟨*names*⟩}

```
\NewDocumentCommand{\settitlemarks}{s m}{}
```

\headrule

```
\newcommand*{\headrule}{}
```

\footrule

```
\newcommand*{\footrule}{}
```

\setheadrule {⟨*length*⟩}

```
\newcommand*{\setheadrule}[1]{}
```

\setfootrule {⟨*length*⟩}

```
\newcommand*{\setfootrule}[1]{}
```

\makeheadrule

```
11 \newcommand*{\makeheadrule}{}
```

\makefootrule

```
12 \newcommand*{\makefootrule}{}
```

\setmarkboth {⟨*code*⟩}

```
13 \newcommand{\setmarkboth}[1]{}
```

\widenhead

```
14 \NewDocumentCommand{\widenhead}{s o o m m}{}
```

\bottitlemarks

```
15 \newcommand*{\bottitlemarks}{}
```

\toptitlemarks

```
16 \newcommand*{\toptitlemarks}{}
```

\firsttitlemarks

```
17 \newcommand*{\firsttitlemarks}{}
```

\nexttitlemarks

```
18 \newcommand*{\nexttoptitlemarks}{}
```

\outertitlemarks

```
19 \newcommand*{\outertitlemarks}{}
```

\innertitlemarks

```
20 \newcommand*{\innertitlemarks}{}
```

\newtitlemark * {⟨*name*⟩}

```
21 \NewDocumentCommand{\newtitlemark}{s m}{}
```

\pretitlemark * {⟨*section*⟩} {⟨*text*⟩}

```
22 \NewDocumentCommand{\pretitlemark}{s m m}{}
```

\ifsamemark {⟨*group*⟩} {⟨*command*⟩} {⟨*true*⟩} {⟨*false*⟩}

```
23 \newcommand{\ifsamemark}[4]{}
```

\setfloathead * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
24 \NewDocumentCommand{\setfloathead}{s o o o m m m m m}{}
```

\setfloatfoot * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
25 \NewDocumentCommand{\setfloatfoot}{s o o o m m m m m}{}
```

\nextfloathead * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
26 \NewDocumentCommand{\nextfloathead}{s o o o m m m m m}{}
```

\nextfloatfoot * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
27 \NewDocumentCommand{\nextfloatfoot}{s o o o m m m m m}{}
```

\newmarkset {⟨*markset*⟩}

```
28 \newcommand{\newmarkset}[1]{}
```

\newextramark * {⟨*markset*⟩} {⟨*macro-name*⟩}

```
29 \NewDocumentCommand{\newextramarkset}{s m m}{}
```

\botextramarks {⟨*markset*⟩}

```
30 \newcommand{\botextramarks}[1]{}
```

\topextramarks {⟨*markset*⟩}

```
31 \newcommand{\topextramarks}[1]{}
```

\firstextramarks {⟨*markset*⟩}

```
32 \newcommand{\firstextramarks}[1]{}
```

\nextextramarks {⟨*markset*⟩}

```
33 \newcommand{\nexttopextramarks}[1]{}
```

\outerextramarks {⟨*markset*⟩}

```
34 \newcommand{\outerextramarks}[1]{}
```

\innerextramarks {⟨*markset*⟩}

```
35 \newcommand{\innerextramarks}[1]{}
```

File 514 **lwarp-titleref.sty**

§ 623 Package

titleref

titleref (*Pkg*) titleref is emulated.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{titleref}[2001/04/05]
2
3 \LetLtxMacro\titleref\nameref
4
5 \providecounter{LWR@currenttitle}
6
7 \newcommand*{\currenttitle}{%
```

```
    \addtocounter{LWR@currenttitle}{1}%
    \label{currenttitle\arabic{LWR@currenttitle}}%
    \nameref{currenttitle\arabic{LWR@currenttitle}}%
}

\newcommand*{\theTitleReference}[2]{}
```

File 515 **lwarp-titlesec.sty**

§ 624 Package

titlesec

(*Emulates or patches code by* Javier Bezos.)

titlesec (*Pkg*) titlesec is emulated. All user options and macros are ignored and disabled.

Discard all options for lwarp-titlesec:

for HTML output:

```
\PackageInfo{lwarp}{Using the lwarp version of package 'titlesec'.}%
\ProvidesPackage{lwarp-titlesec}[2016/03/21]

\newbool{LWR@loadtitleps}
\boolfalse{LWR@loadtitleps}

\DeclareOption{pagestyles}{
    \booltrue{LWR@loadtitleps}
}

\DeclareOption*{}

\ProcessOptions\relax

\ifbool{LWR@loadtitleps}{
    \RequirePackage{lwarp-titleps}
}{}
```

\titlelabel {⟨*label-format*⟩}

```
\newcommand*{\titlelabel}[1]{}
```

\titleformat* {⟨*command*⟩} {⟨*format*⟩}

\titleformat {⟨*command*⟩} [⟨*shape*⟩] {⟨*format*⟩} {⟨*label*⟩} {⟨*sep*⟩} {⟨*begfore*⟩} [⟨*after*⟩]

```
\newcommand\titleformat{%
  \@ifstar{\ttl@format@s}%
          {\ttl@format@i}}
\newcommand{\ttl@format@s}[1]{}
\NewDocumentCommand{\ttl@format@i}{m o m m m m o}{}
```

\chaptertitlename

```
\@ifundefined{@chapapp}{\let\@chapapp\chaptername}{}
\newcommand\chaptertitlename{\@chapapp}
```

\titlespacing * {⟨*command*⟩} {⟨*left*⟩} {⟨*before*⟩} {⟨*after*⟩} [⟨*right*⟩]

```
26 \NewDocumentCommand{\titlespacing}{s m m m m o}{}
```

\filright

```
27 \newcommand*{\filright}{}
```

\filcenter

```
28 \newcommand*{\filcenter}{}
```

\filleft

```
29 \newcommand*{\filleft}{}
```

\fillast

```
30 \newcommand*{\fillast}{}
```

\filinner

```
31 \newcommand*{\filinner}{}
```

\filouter

```
32 \newcommand*{\filouter}{}
```

\wordsep

```
33 \newcommand\wordsep{\fontdimen\tw@\font \@plus
34   \fontdimen\thr@@\font \@minus \fontdimen4\font}
```

\titleline * [⟨*align*⟩] {⟨*material*⟩}

```
35 \NewDocumentCommand{\titleline}{s o m}{}
```

\titlerule [⟨*height*⟩]

```
36 \providecommand*\titlerule{\@ifstar{\ttl@row}{\ttl@rule}}
37 \newcommand*{\ttl@rule}[1][]{}
38 \newcommand*{\ttl@row}[2][]{}
```

\iftitlemeasuring {⟨*true*⟩} {⟨*false*⟩}

```
39 \newcommand{\iftitlemeasuring}[2]{#2}
```

\assignpagestyle {⟨*command*⟩} {⟨*pagestyle*⟩}

```
40 \newcommand{\assignpagestyle}[2]{#2}
```

\titleclass {⟨*name*⟩} [⟨*startlevel*⟩] {⟨*class*⟩} [⟨*cmd*⟩]

```
41 \NewDocumentCommand{\titleclass}{m o m o}{}
```

File 516 **lwarp-titletoc.sty**

§ 625 Package

titletoc

(*Emulates or patches code by* Javier Bezos.)

titletoc (*Pkg*) titletoc is emulated. All user options and macros are ignored and disabled.

Discard all options for lwarp-titletoc:

for HTML output:

```
\LWR@ProvidesPackageDrop{titletoc}[2011/12/15]
```

\dottedcontents {⟨*section*⟩} [⟨*left*⟩] {⟨*above*⟩} {⟨*label*⟩} {⟨*leader*⟩}

```
\NewDocumentCommand{\dottedcontents}{m o m m m}{}
```

\titlecontents * {⟨*section*⟩} [⟨*left*⟩] {⟨*above*⟩} {⟨*numbered*⟩} {⟨*numberless*⟩} {⟨*filler*⟩} [⟨*below or begin*⟩] [⟨*separator*⟩] [⟨*end*⟩]

```
\newcommand{\titlecontents}{\@ifstar{\ttl@tcstar}{\ttl@tcnostar}}
\NewDocumentCommand{\ttl@tcstar}{m o m m m m o o o}{}
\NewDocumentCommand{\ttl@tcnostar}{m o m m m m o}{}
```

\contentsmargin [⟨*correction*⟩] {⟨*right*⟩}

```
\newcommand{\contentsmargin}[2][]{}
```

\thecontentslabel

```
\newcommand*{\thecontentslabel}{thecontentslabel}
```

\thecontentspage

```
\newcommand*{\thecontentspage}{thecontentspage}
```

\contentslabel [⟨*format*⟩] {⟨*space*⟩}

```
\newcommand{\contentslabel}[2][]{\thecontentslabel}
```

\contentspage [⟨*format*⟩]

```
\newcommand{\contentspage}[1][]{\thecontentspage}
```

\contentspush {⟨*text*⟩}

```
\newcommand{\contentspush}[1]{}
```

\contentsuse {⟨*name*⟩} {⟨*text*⟩}

```
\newcommand{\contentsuse}[2]{}
```

\startcontents [⟨*name*⟩]

```
\newcommand*{\startcontents}[1][]{}
```

\stopcontents [⟨*name*⟩]

```
\newcommand*{\stopcontents}[1][]{}
```

\resumecontents [⟨*name*⟩]

```
\newcommand*{\resumecontents}[1][]{}
```

\printcontents [⟨*name*⟩] {⟨*prefix*⟩} {⟨*start*⟩} {⟨*code*⟩}

```
\newcommand{\printcontents}[4][]{}
```

\startlist [⟨*name*⟩] {⟨*list*⟩}

```
\newcommand{\startlist}[2][]{}
```

\stoplist [⟨*name*⟩] {⟨*list*⟩}

```
\newcommand{\stoplist}[2][]{}
```

\resumelist [⟨*name*⟩] {⟨*list*⟩}

```
\newcommand{\resumelist}[2][]{}
```

\printlist [⟨*name*⟩] {⟨*list*⟩} {⟨*prefix*⟩} {⟨*code*⟩}

```
\newcommand{\printlist}[4][]{}
```

File 517 **lwarp-titling.sty**

§ 626 Package

titling

(*Emulates or patches code by* Peter Wilson.)

titling (*Pkg*)

package support
⚠ load order

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 69.8.

The various titling footnote restyling commands have no effect.

Pass all options to lwarp-titling:

for HTML output:

```
\LWR@ProvidesPackagePass{titling}[2009/09/04]
```

\@bsmtitlempty

Patch \@bsmtitlempty:

```
\let\LWR@orig@bsmtitlempty\@bsmtitlempty
\renewcommand*{\@bsmtitlempty}{%
\LWR@orig@bsmtitlempty%
}
```

\keepthetitle Patch \keepthetitle:

```
\let\LWR@origkeepthetitle\keepthetitle
\renewcommand*{\keepthetitle}{%
\LWR@orig@keepthetitle%
}
```

\killtitle Patch \killtitle:

```
\let\LWR@origkilltitle\killtitle
\renewcommand*{\killtitle}{%
\LWR@orig@killtitle%
}
```

titlingpage (*env.*)

```
\renewenvironment*{titlingpage}
{%
```

Start an HTML titlepage div:

```
\LWR@printpendingfootnotes
\begin{titlepage}
```

Prepare for a custom version of \maketitle inside the titlingpage:

```
\LWR@maketitlesetup
\let\maketitle\LWR@titlingmaketitle
}
{
```

At the end of the environment, end the HTML titlepage div:

```
\end{titlepage}
}
```

Patch the pre/post title/author/date to add HTML tags, then initilize:

```
\AtBeginDocument{
    \pretitle{}
    \posttitle{}

    \preauthor{}
    \postauthor{}

    \predate{}
    \postdate{}
}
```

\LWR@maketitlesetup Patches \thanks macros.

```
\renewcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
    \def\@makefnmark{\textsuperscript{\@thefnmark}}%
```

\thefootnote ⇒ \nameuse{arabic}{footnote}, or
\thefootnote ⇒ \nameuse{fnsymbol}{footnote}

Redefine the footnote text:

```
\long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
    \makethanksmark~%
```

\makethanksmark ⇒ \thanksfootmark ⇒ \tamark ⇒
\@thefnmark ⇒ \itshape a (or similar)

Print the text:

```
    {##1}%
}% \@makefntext
}
```

\thanksfootmark

```
\renewcommand{\thanksfootmark}{%
%   \hb@xt@\thanksmarkwidth{\hfil\normalfont%
        \thanksscript{%
            \thanksfootpre \tamark \thanksfootpost%
        }%
%   }%
}
```

\maketitle HTML mode. Creates an HTML titlepage div and typesets the title, etc.

Code from the titling package is adapted, simplified, and modified for HTML output.

```
\renewcommand*{\maketitle}{%
```

An HTML titlepage is used for all classes.

```
    \begin{titlepage}
```

Select which kind of footnote marks to use:

```
    \@bsmarkseries
```

Set up special patches:

```
    \LWR@maketitlesetup
```

Typeset the title, etc:

```
    \@maketitle
```

Immediately generate any \thanks footnotes:

```
    \LWR@stoppars\@thanks\LWR@startpars
```

Close the HTML titlepage div:

```
\end{titlepage}
```

Reset the footnote counter:

```
    \@bscontmark
}
```

`\@maketitle` Typesets the title, etc. Patched for HTML.

```
\providecommand*{\@maketitle}{}
\renewrobustcmd{\@maketitle}{%
    \maketitlehooka
    {
        \LWR@stoppars\LWR@htmltag{\LWR@tagtitle}%
        \@bspretitle \@title \@bsposttitle%
        \LWR@htmltag{\LWR@tagtitleend}\LWR@startpars%
    }
    \maketitlehookb
    {
        \begin{BlockClass}{author}
        \renewcommand{\and}{%
            \end{BlockClass}%
            \begin{BlockClass}{oneauthor}%
        }
        \begin{BlockClass}{oneauthor}%
        \@bspreauthor \@author  \@bspostauthor%
        \end{BlockClass}%
        \end{BlockClass}%
    }
    \maketitlehookc
    {
        \begin{BlockClass}{titledate}%
        \@bspredate \@date \@bspostdate%
        \end{BlockClass}%
    }
    \maketitlehookd
}
```

`\LWR@titlingmaketitle` `\maketitle` for use inside an HTML `titlingpage` environment.

```
\renewcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
\LWR@stoppars\@thanks\LWR@startpars
```

Select which kind of footnote marks to use:

```
\@bsmarkseries
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any \thanks footnotes:

```
\LWR@stoppars\@thanks\LWR@startpars
```

Reset the footnote counter:

```
\@bscontmark
}
```

\thanksmarkseries {⟨*series*⟩}

Sets the type of footnote marks used by \thanks, where type is 'arabic', 'roman', 'fnsymbol', etc.

```
\renewcommand{\thanksmarkseries}[1]{%
\def\@bsmarkseries{\renewcommand{\thefootnote}{\@nameuse{#1}{footnote}}}%
}
```

Set default titlepage thanks footnote marks. See section 69.7.

```
\IfClassLoadedTF{memoir}{
   \thanksmarkseries{arabic}
}{% not memoir
\if@titlepage
   \thanksmarkseries{arabic}
\else
   \thanksmarkseries{fnsymbol}
\fi
}% not memoir
```

File 518 **lwarp-tocbasic.sty**

§ 627 Package

tocbasic

(*Emulates or patches code by* Markus Kohm.)

tocbasic (*Pkg*) tocbasic is nullified for lwarp.

This package may be loaded standalone, but is also loaded automatically if koma-script classes are in use. \DeclareDocumentCommand is used to overwrite the koma-script definitions.

for HTML output:

```
\LWR@ProvidesPackagePass{tocbasic}[2018/12/30]

\DeclareDocumentCommand{\usetocbasicnumberline}{o}{}
\DeclareDocumentCommand{\DeclareTOCStyleEntry}{o m m}{}
\DeclareDocumentCommand{\DeclareTOCStyleEntries}{o m m}{}
\DeclareDocumentCommand{\DeclareTOCEntryStyle}{m o m}{}
\DeclareDocumentCommand{\DefineTOCEntryOption}{m o m}{}
\DeclareDocumentCommand{\DefineTOCEntryBooleanOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryCommandOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryIfOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryLengthOption}{m o m m m}{}
```

```
\DeclareDocumentCommand{\DefineTOCEntryNumberOption}{m o m m m}{}
\DeclareDocumentCommand{\CloneTOCEntryStyle}{m m}{}
\DeclareDocumentCommand{\TOCEntryStyleInitCode}{m m}{}
\DeclareDocumentCommand{\TOCEntryStyleStartInitCode}{m m}{}
```

File 519 **lwarp-tocbibind.sty**

§ 628 Package

tocbibind

(*Emulates or patches code by* Peter Wilson.)

tocbibind (*Pkg*) tocbibind is patched for use by lwarp.

placement and TOC options An index may be placed inline with other HTML text, or on its own HTML page:

makeidx (*Pkg*) **Inline, with a manual TOC entry:**

A commonly-used method to introduce an index in a LaTeX document:

```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\printindex
```

makeidx (*Pkg*) **On its own HTML page, with a manual TOC entry:**

```
\begin{warpprint}
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\end{warpprint}
\ForceHTMLPage
\ForceHTMLTOC
\printindex
```

tocbibind (*Pkg*) **Inline, with an automatic TOC entry:**

The tocbibind package may be used to automatically place an entry in the TOC.

```
\usepackage[nottoc]{tocbibind}
. . .
\cleardoublepage
\phantomsection % to fix print-version index link
\printindex
```

tocbibind (*Pkg*) **On its own HTML page, with an automatic TOC entry:**

```
\usepackage[nottoc]{tocbibind}
. . .
\cleardoublepage
\phantomsection % to fix print-version index link
\ForceHTMLPage
\printindex
```

numindex (*Opt*) [tocbibind] Use the tocbibind numindex option to generate a numbered index. Without this option, the index heading has no number.

numbered index section

Other packages, such as imakeidx, may also have options for including the index in the Table of Contents.

for HTML output:

```
\let\simplechapterdelim\relax

\LWR@ProvidesPackagePass{tocbibind}[2010/10/13]

\renewenvironment{theindex}%
{%
    \if@bibchapter
       \if@donumindex
           \chapter{\indexname}
       \else
         \if@dotocind
           \chapter*{\indexname}
           \addcontentsline{toc}{chapter}{\LWR@isolate{\indexname}}
         \else
           \chapter*{\indexname}
         \fi
       \fi
    \else
       \if@donumindex
           \section{\indexname}
       \else
         \if@dotocind
           \section*{\indexname}
           \addcontentsline{toc}{\@tocextra}{\LWR@isolate{\indexname}}
         \else
           \section*{\indexname}
         \fi
       \fi
    \fi
\LetLtxMacro\item\LWR@indexitem%
\LetLtxMacro\subitem\LWR@indexsubitem%
\LetLtxMacro\subsubitem\LWR@indexsubsubitem%
}{}
```

The following code is shared by anonchap.

```
\DeclareDocumentCommand{\simplechapter}{O{\@empty}}{%
    \def\@chapcntformat##1{%
        #1~\csname the##1\endcsname\simplechapterdelim\quad%
    }%
}

\DeclareDocumentCommand{\restorechapter}{}{%
\let\@chapcntformat\@seccntformat%
}
```

File 520 **lwarp-tocdata.sty**

§ 629 Package tocdata

(*Emulates or patches code by* Brian Dunn.)

tocdata (*Pkg*) tocdata is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{tocdata}[2019/07/06]
```

```
\renewcommand*{\LWR@maybetocdata}{%
    \ifdefempty{\TD@thistocdata}{}{%
        \qquad \InlineClass{authorartist}{\tocdataformat{\TD@thistocdata}}%
        \def\TD@thistocdata{}
    }
}

\renewrobustcmd{\tocdatapartprint}[4]
{%
    \InlineClass{authorartist}{%
        \qquad --- %
        \TDoptionalnameprint{#1}\TDoptionalnameprint{#2}#3#4%
    }%
}

\@ifundefined{chapter}{}{
    \let\tocdatachapterprint\tocdatapartprint
}
\let\tocdatasectionprint\tocdatapartprint
\let\tocdatasubsectionprint\tocdatapartprint

\newcommand*{\LWR@TD@settextalign}[1]{%
    \def\LWR@TD@textalign{justify}%
    \ifcsstring{TD@#1align}{\centering}%
        {\def\LWR@TD@textalign{center}}%
        {}%
    \ifcsstring{TD@#1align}{\raggedleft}%
        {\def\LWR@TD@textalign{right}}%
        {}%
    \ifcsstring{TD@#1align}{\raggedright}%
        {\def\LWR@TD@textalign{left}}%
        {}%
}

\renewcommand{\TDartistauthorprint}[5]{%
    \LWR@TD@settextalign{#1}%
    \begin{BlockClass}[text-align:\LWR@TD@textalign]{floatnotes}%
   \InlineClass{authorartist}{\TDoptionalnameprint{#2}\TDoptionalnameprint{#3}#4#5}%
    \end{BlockClass}%
}

\newcommand*{\LWR@TD@setnamealign}[1]{%
    \def\LWR@TD@textalign{justify}%
    \ifcsstring{TD@#1textalign}{\centering}%
        {\def\LWR@TD@textalign{center}}%
        {}%
    \ifcsstring{TD@#1textalign}{\raggedleft}%
        {\def\LWR@TD@textalign{right}}%
        {}%
    \ifcsstring{TD@#1textalign}{\raggedright}%
        {\def\LWR@TD@textalign{left}}%
        {}%
}

\renewcommand{\TDartistauthortextprint}[2]{%
    \LWR@TD@setnamealign{#1}%
    \begin{BlockClass}[text-align:\LWR@TD@textalign]{floatnotes}%
    #2%
    \end{BlockClass}%
}
```

File 521 **lwarp-tocenter.sty**

§ 630 Package

tocenter

tocenter (*Pkg*) tocenter is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tocenter}[2004/12/09]

\NewDocumentCommand{\ToCenter}{s o m m}{}
\NewDocumentCommand{\FromMargins}{s o m m m m}{}
```

File 522 **lwarp-tocloft.sty**

§ 631 Package

tocloft

(*Emulates or patches code by* Peter Wilson.)

tocloft (*Pkg*) tocloft is emulated. Most user options and macros are ignored and disabled. \newlistof and \cftchapterprecis are supported.

tocloft (*Pkg*) ⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its titles option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

Discard all options for lwarp-tocloft:

for HTML output:

```
\LWR@ProvidesPackageDrop{tocloft}[2017/08/31]
```

\tocloftpagestyle {⟨*style*⟩}

```
\newcommand{\tocloftpagestyle}[1]{}
```

\cftmarktoc

```
\newcommand*{\cftmarktoc}{}
```

\cfttoctitlefont

```
\newcommand*{\cfttoctitlefont}{}
```

\cftaftertoctitle

```
\newcommand*{\cftaftertoctitle}{}

\newlength{\cftbeforetoctitleskip}
\newlength{\cftaftertoctitleskip}
```

\cftmarklof

```
\newcommand*{\cftmarklof}{}
```

\cftloftitlefont

```
\newcommand*{\cftloftitlefont}{}
```

\cftafterloftitle

```
\newcommand*{\cftafterloftitle}{}

\newlength{\cftbeforeloftitleskip}
\newlength{\cftafterloftitleskip}
```

\cftmarklot

```
\newcommand*{\cftmarklot}{}
```

\cftlottitlefont

```
\newcommand*{\cftlottitlefont}{}
```

\cftafterlottitle

```
\newcommand*{\cftafterlottitle}{}

\newlength{\cftbeforelottitleskip}
\newlength{\cftafterlottitleskip}
```

\cftdot

```
\providecommand*{\cftdot}{.}
```

\cftdotsep

```
\providecommand*{\cftdotsep}{1}
```

\cftnodots

```
\providecommand*{\cftnodots}{5000}
```

\cftdotfill {⟨*sep*⟩}

```
\providecommand{\cftdotfill}[1]{}
```

\cftsetpnumwidth {⟨*length*⟩}

```
\DeclareDocumentCommand{\cftsetpnumwidth}{m}{}
```

\cftsetrmarg {⟨*length*⟩}

```
\DeclareDocumentCommand{\cftsetrmarg}{m}{}
```

\cftpnumalign {⟨*alignment*⟩}

```
\DeclareDocumentCommand{\cftpnumalign}{m}{}

\LWR@providelength{\cftparskip}
```

The part-related items are also provided by memoir:

```
\LWR@providelength{\cftbeforepartskip}
\LWR@providelength{\cftpartindent}
\LWR@providelength{\cftpartnumwidth}
\providecommand*{\cftpartfont}{}
\providecommand*{\cftpartpresnum}{}
\providecommand*{\cftpartaftersnum}{}
\providecommand*{\cftpartaftersnumb}{}
\providecommand*{\cftpartleader}{}
\providecommand*{\cftpartdotsep}{1}
\providecommand*{\cftpartpagefont}{}
\providecommand*{\cftpartafterpnum}{}
```

memoir uses the full name “chapter” instead of “chap”:

```
\LWR@providelength{\cftbeforechapskip}
\LWR@providelength{\cftchapindent}
\LWR@providelength{\cftchapnumwidth}
\newcommand*{\cftchapfont}{}
\newcommand*{\cftchappresnum}{}
\newcommand*{\cftchapaftersnum}{}
\newcommand*{\cftchapaftersnumb}{}
\newcommand*{\cftchapleader}{}
\newcommand*{\cftchapdotsep}{1}
\newcommand*{\cftchappagefont}{}
\newcommand*{\cftchapafterpnum}{}
```

The following do not appear in memoir:

```
\LWR@providelength{\cftbeforesecskip}
\LWR@providelength{\cftsecindent}
\LWR@providelength{\cftsecnumwidth}
\newcommand*{\cftsecfont}{}
\newcommand*{\cftsecpresnum}{}
\newcommand*{\cftsecaftersnum}{}
\newcommand*{\cftsecaftersnumb}{}
\newcommand*{\cftsecleader}{}
\newcommand*{\cftsecdotsep}{1}
\newcommand*{\cftsecpagefont}{}
\newcommand*{\cftsecafterpnum}{}

\LWR@providelength{\cftbeforesubsecskip}
\LWR@providelength{\cftsubsecindent}
\LWR@providelength{\cftsubsecnumwidth}
\newcommand*{\cftsubsecfont}{}
\newcommand*{\cftsubsecpresnum}{}
\newcommand*{\cftsubsecaftersnum}{}
\newcommand*{\cftsubsecaftersnumb}{}
\newcommand*{\cftsubsecleader}{}
\newcommand*{\cftsubsecdotsep}{1}
\newcommand*{\cftsubsecpagefont}{}
\newcommand*{\cftsubsecafterpnum}{}

\LWR@providelength{\cftbeforesubsubsecskip}
\LWR@providelength{\cftsubsubsecindent}
\LWR@providelength{\cftsubsubsecnumwidth}
\newcommand*{\cftsubsubsecfont}{}
\newcommand*{\cftsubsubsecpresnum}{}
```

```
\newcommand*{\cftsubsubsecaftersnum}{}
\newcommand*{\cftsubsubsecaftersnumb}{}
\newcommand*{\cftsubsubsecleader}{}
\newcommand*{\cftsubsubsecdotsep}{1}
\newcommand*{\cftsubsubsecpagefont}{}
\newcommand*{\cftsubsubsecafterpnum}{}

\LWR@providelength{\cftbeforeparaskip}
\LWR@providelength{\cftparaindent}
\LWR@providelength{\cftparanumwidth}
\newcommand*{\cftparafont}{}
\newcommand*{\cftparapresnum}{}
\newcommand*{\cftparaaftersnum}{}
\newcommand*{\cftparaaftersnumb}{}
\newcommand*{\cftparaleader}{}
\newcommand*{\cftparadotsep}{1}
\newcommand*{\cftparapagefont}{}
\newcommand*{\cftparaafterpnum}{}

\LWR@providelength{\cftbeforesubparaskip}
\LWR@providelength{\cftsubparaindent}
\LWR@providelength{\cftsubparanumwidth}
\newcommand*{\cftsubparafont}{}
\newcommand*{\cftsubparapresnum}{}
\newcommand*{\cftsubparaaftersnum}{}
\newcommand*{\cftsubparaaftersnumb}{}
\newcommand*{\cftsubparaleader}{}
\newcommand*{\cftsubparadotsep}{1}
\newcommand*{\cftsubparapagefont}{}
\newcommand*{\cftsubparaafterpnum}{}

\LWR@providelength{\cftbeforefigskip}
\LWR@providelength{\cftfigindent}
\LWR@providelength{\cftfignumwidth}
\newcommand*{\cftfigfont}{}
\newcommand*{\cftfigpresnum}{}
\newcommand*{\cftfigaftersnum}{}
\newcommand*{\cftfigaftersnumb}{}
\newcommand*{\cftfigleader}{}
\newcommand*{\cftfigdotsep}{1}
\newcommand*{\cftfigpagefont}{}
\newcommand*{\cftfigafterpnum}{}

\LWR@providelength{\cftbeforesubfigskip}
\LWR@providelength{\cftsubfigindent}
\LWR@providelength{\cftsubfignumwidth}
\newcommand*{\cftsubfigfont}{}
\newcommand*{\cftsubfigpresnum}{}
\newcommand*{\cftsubfigaftersnum}{}
\newcommand*{\cftsubfigaftersnumb}{}
\newcommand*{\cftsubfigleader}{}
\newcommand*{\cftsubfigdotsep}{1}
\newcommand*{\cftsubfigpagefont}{}
\newcommand*{\cftsubfigafterpnum}{}

\LWR@providelength{\cftbeforetabskip}
\LWR@providelength{\cfttabindent}
\LWR@providelength{\cfttabnumwidth}
\newcommand*{\cfttabfont}{}
```

```
\newcommand*{\cfttabpresnum}{}
\newcommand*{\cfttabaftersnum}{}
\newcommand*{\cfttabaftersnumb}{}
\newcommand*{\cfttableader}{}
\newcommand*{\cfttabdotsep}{1}
\newcommand*{\cfttabpagefont}{}
\newcommand*{\cfttabafterpnum}{}

\LWR@providelength{\cftbeforesubtabskip}
\LWR@providelength{\cftsubtabindent}
\LWR@providelength{\cftsubtabnumwidth}
\newcommand*{\cftsubtabfont}{}
\newcommand*{\cftsubtabpresnum}{}
\newcommand*{\cftsubtabaftersnum}{}
\newcommand*{\cftsubtabaftersnumb}{}
\newcommand*{\cftsubtableader}{}
\newcommand*{\cftsubtabdotsep}{1}
\newcommand*{\cftsubtabpagefont}{}
\newcommand*{\cftsubtabafterpnum}{}

\DeclareDocumentCommand{\cftsetindents}{m m m}{}

\providecommand{\cftpagenumbersoff}[1]{}
\providecommand{\cftpagenumberson}[1]{}
```

\newlistentry [⟨*within*⟩] {⟨*counter*⟩} {⟨*ext*⟩} {⟨*level-1*⟩}

```
\DeclareDocumentCommand{\newlistentry}{o m m m}
{%
\LWR@traceinfo{newlistentry #2 #3 #4}%
\IfValueTF{#1}%
{%
    \@ifundefined{c@#2}{%
        \newcounter{#2}[#1]%
        \expandafter\edef\csname the#2\endcsname{%
          \expandafter\noexpand\csname the#1\endcsname.\noexpand\arabic{#2}%
        }%
    }{}%
}%
{%
    \@ifundefined{c@#2}{%
        \newcounter{#2}%
    }{}%
}%
\@namedef{l@#2}##1##2{%
    \hypertocfloat{1}{#2}{#3}{##1}{##2}%
    \def\cftwhatismyname{#2}% from memoir
}%
\expandafter\newlength\csname cftbefore#2skip\endcsname%
\expandafter\newlength\csname cft#2indent\endcsname%
\expandafter\newlength\csname cft#2numwidth\endcsname%
\@namedef{cft#2font}{}%
\@namedef{cft#2presnum}{}%
\@namedef{cft#2aftersnum}{}%
\@namedef{cft#2aftersnumb}{}%
\@namedef{cft#2leader}{}%
\@namedef{cft#2dotsep}{1}%
\@namedef{cft#2pagefont}{}%
```

```
\@namedef{cft#2afterpnum}{}%
\@namedef{toclevel@#2}{#4}%
\@namedef{cft#2fillnum}##1{}%
\LWR@traceinfo{newlistentry done}%
}
```

`\newlistof` [⟨*within*⟩] {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the `\newfloat` mechanism.

```
\DeclareDocumentCommand{\newlistof}{o m m m}
{%
    \IfValueTF{#1}%
        {\newlistentry[#1]{#2}{#3}{0}}%
        {\newlistentry{#2}{#3}{0}}%
    \@namedef{ext@#2}{#3}%
    \@ifundefined{c@#3depth}{\newcounter{#3depth}}{}%
    \setcounter{#3depth}{1}%
    \@namedef{cftmark#3}{}%
    \@namedef{listof#2}{\LWR@listof{#2}{#4}}%
    \@namedef{@cftmake#3title}{}%
    \expandafter\newlength\csname cftbefore#3titleskip\endcsname%
    \expandafter\newlength\csname cftafter#3titleskip\endcsname%
    \@namedef{cft#3titlefont}{}%
    \@namedef{cftafter#3title}{}%
    \@namedef{cft#3prehook}{}%
    \@namedef{cft#3posthook}{}%
}
```

`\cftchapterprecis` {⟨*text*⟩}

```
\newcommand{\cftchapterprecis}[1]{%
  \cftchapterprecishere{#1}
  \cftchapterprecistoc{#1}}
\newcommand{\cftchapterprecishere}[1]{%
  \begin{quote}\textit{#1}\end{quote}}
\newcommand{\cftchapterprecistoc}[1]{
  \addtocontents{toc}{%
  {
     \protect\begin{quote}#1\protect\end{quote}}
  }
}
```

File 523 **lwarp-tocstyle.sty**

§ 632 Package

tocstyle

`tocstyle` (*Pkg*) tocstyle is ignored.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{tocstyle}[2017/02/23]

\newcommand*{\usetocstyle}[2][]{}
\newcommand*{\deactivatetocstyle}[1][]{}
\newcommand*{\reactivatetocstyle}[1][]{}
\NewDocumentCommand{\settocfeature}{o o m m}{}
```

```
\NewDocumentCommand{\settocstylefeature}{o m m}{}
\NewDocumentCommand{\newtocstyle}{o o m m}{}
\newcommand*{\aliastoc}[2]{}
\newcommand*{\showtoc}[2][]{}
\newcommand{\iftochasdepth}[4]{}
```

File 524 **lwarp-todo.sty**

§ 633 Package todo

(*Emulates or patches code by* Federico Garcia.)

todo (*Pkg*) todo is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{todo}[2010/03/31]

\renewcommand\todoitem[2]{%
    \refstepcounter{todo}%
    \item[%
        \HTMLunicode{2610} \quad%
        \ref{todopage:\thetodo}
     ] : {\todoformat\ifx#1\todomark\else\textbf{#1} \fi}#2%
    \label{todolbl:\thetodo}%
}%

\renewcommand\doneitem[2]{%
    \stepcounter{todo}%
    \item[%
        \HTMLunicode{2611} \quad%
        \ref{todopage:\thetodo}
    ] \@nameuse{@done\the\c@todo}:
        {\todoformat\ifx#1\todomark\else\textbf{#1} \fi}#2%
}
```

The following are not errors because the code will still compile and be usable if the patch is not possible.

```
\xpatchcmd{\@displaytodo}
    {\todoformat #1}{\todoformat \textbf{#1}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch @displaytodo.}}

\xpatchcmd{\@displayfulltodo}
    {\todoformat #1}{\todoformat \textbf{#1}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch @displayfulltodo.}}

\patchcmd{\todoenv}{\itshape see text.}{\textit{see text.}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch todoenv.}}

\patchcmd{\astodos}{\todoformat #1}{\todoformat \textbf{#1}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch astodos.}}
```

If cleveref is in use, name the new todo notes:

```
\AtBeginDocument{
\ifdef{\crefname}{
    \crefname{todo}{todo}{todos}
```

```
    \Crefname{todo}{Todo}{Todos}
}{}
}
```

File 525 **lwarp-todonotes.sty**

§ 634 Package **todonotes**

(*Emulates or patches code by* Henrik Skov Midtiby.)

todonotes (*Pkg*) todonotes is emulated.

The documentation for todonotes and luatodonotes have an example with a todo inside a caption. If this example does not work it will be necessary to move the todo outside of the caption.

for HTML output:

```
\LWR@ProvidesPackagePass{todonotes}[2012/07/25]

\if@todonotes@disabled
\else

\newcommand{\ext@todo}{tdo}

\renewcommand{\l@todo}[2]{\hypertocfloat{1}{todo}{ldo}{#1}{#2}}

\let\LWRTODONOTES@orig@todototoc\todototoc

\renewcommand*{\todototoc}{%
\LWR@phantomsection%
\LWRTODONOTES@orig@todototoc%
}

\renewcommand{\@todonotes@drawMarginNoteWithLine}{
\fcolorbox
    {\@todonotes@currentbordercolor}
    {\@todonotes@currentbackgroundcolor}
    {\arabic{@todonotes@numberoftodonotes}}
\marginpar{\@todonotes@drawMarginNote}
}

\renewcommand{\@todonotes@drawInlineNote}{%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \if@todonotes@authorgiven%
        {\@todonotes@author:\,}%
        \fi%
        \@todonotes@text%
    }%
}

\renewcommand{\@todonotes@drawMarginNote}{%
    \if@todonotes@authorgiven%
        \@todonotes@author\par%
    \fi%
    \arabic{@todonotes@numberoftodonotes}: %
```

```
    \fcolorbox%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \@todonotes@sizecommand%
        \@todonotes@text %
    }%
}%

\renewcommand{\@todonotes@drawLineToRightMargin}{}

\renewcommand{\@todonotes@drawLineToLeftMargin}{}

\renewcommand{\missingfigure}[2][]{%
\setkeys{todonotes}{#1}%
\addcontentsline{tdo}{todo}{\@todonotes@MissingFigureText: #2}%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentfigcolor}%
    {%
        \setlength{\fboxrule}{4pt}%
        \fcolorbox{red}{white}{Missing figure} \quad #2%
    }
}

\LetLtxMacro\LWRTODONOTES@orig@todo\@todo

\RenewDocumentCommand{\@todo}{o m}{%
\begingroup%
\renewcommand*{\phantomsection}{}%
\IfValueTF{#1}{%
    \LWRTODONOTES@orig@todo[#1]{#2}%
}{%
    \LWRTODONOTES@orig@todo{#2}%
}
\endgroup%
}

\fi% \if@todonotes@disabled
```

File 526 **lwarp-topcapt.sty**

§ 635 Package topcapt

topcapt (*Pkg*) topcapt is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{topcapt}[2004/12/11]

\LetLtxMacro\topcaption\caption
```

File 527 **lwarp-tram.sty**

§ 636 Package tram

tram (*Pkg*) tram is emulated.

⚠ block only The HTML emulation uses a , which must not appear inside an HTML or an HTML paragraph. For this reason, the tram environment should only be used to contain paragraphs inside a \parbox or minipage. tram should not be used to mark up inline text.

To disable tram, allowing source compatibility with inline uses:

```
\begin{warpHTML}
\renewenvironment{tram}[1][]{}{}
\end{warpHTML}
```

for HTML output:

```
\LWR@ProvidesPackageDrop{tram}[2013/04/04]

\newenvironment{tram}[1][]%
    {\BlockClass[background:lightgray]{tram}}
    {\endBlockClass}
```

File 528 **lwarp-transparent.sty**

§ 637 Package transparent

(*Emulates or patches code by* Heiko Oberdiek.)

transparent (*Pkg*) transparent is emulated. \texttransparent works for inline objects. \transparent only works for \includegraphics.

⚠ Not XeLaTeX! Note that transparent does not work with XeLaTeX.

for HTML output:

```
\LWR@ProvidesPackagePass{transparent}[2019/11/29]

\newcommand*{\LWR@HTML@transparent}[1]{\edef\LWR@opacity{#1}}

\LWR@formatted{transparent}


\newcommand*{\LWR@HTML@texttransparent}[2]{%
\begingroup%
\transparent{#1}%
\InlineClass[opacity: #1]{transparent}{#2}%
\endgroup%
}

\LWR@formatted{texttransparent}
```

File 529 **lwarp-trimclip.sty**

§ 638 Package trimclip

trimclip (*Pkg*) trimclip is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{trimclip}[2018/04/08]
```

The third argument, the text, is not touched. This allows \bgroup / \egroup, and verbatim content.

```
\csdef{trimbox}{\@ifstar\@gobble\@gobble}
\csletcs{trimbox*}{trimbox}
\def\endtrimbox{}
\csletcs{endtrimbox*}{endtrimbox}

\csletcs{clipbox}{trimbox}
\csletcs{clipbox*}{trimbox}
\csletcs{endclipbox}{endtrimbox}
\csletcs{endclipbox*}{endtrimbox}

\csletcs{marginbox}{trimbox}
\csletcs{marginbox*}{trimbox}
\csletcs{endmarginbox}{endtrimbox}
\csletcs{endmarginbox*}{endtrimbox}
```

File 530 **lwarp-trivfloat.sty**

§ 639 Package

trivfloat

(*Emulates or patches code by* Joseph Wright.)

trivfloat (*Pkg*) trivfloat is forced to use the built-in lwarp emulation for floats.

To create a new float type and change its name:

```
\trivfloat{example}
\renewcommand{\examplename}{Example Name}
\crefname{example}{example}{examples}
\Crefname{example}{Example}{Examples}
```

Discard all options for lwarp-trivfloat. This tells trivfloat not to use floatrow or memoir.

```
\LWR@ProvidesPackageDrop{trivfloat}[2009/04/23]
\LWR@origRequirePackage{trivfloat}
```

\tfl@chapter@fix Nullified at the beginning of the document. Is used by trivfloat to correct float chapter numbers, but is not needed for lwarp.

```
\AtBeginDocument{\DeclareDocumentCommand{\tfl@chapter@fix}{m m}{}}
```

§ 639.1 Combining \newfloat, \trivfloat, and algorithmicx

For both print and HTML output:

⚠ When using float, trivfloat, or algorithmicx at the same time, be aware of conflicting file usage. algorithmicx uses .loa. trivfloat by default starts with .loa and goes up for additional floats, skipping .lof and .lot.

⚠ When using \newfloat, be sure to manually assign higher letters to the \newfloat files to avoid .loa used by algorithmicx, and any files used by trivfloat. Also avoid using .lof and .lot.

⚠ When using \trivfloat, you may force it to avoid conflicting with algorithmicx by starting trivfloat's file extensions with .lob:

```
\makeatletter
\setcounter{tfl@float@cnt}{1} % start trivfloats with .lob
\makeatletter
```

File 531 **lwarp-truncate.sty**

§ 640 Package truncate

truncate (*Pkg*) truncate is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{truncate}[2001/08/20]

\providecommand{\TruncateMarker}{}
\newcommand{\truncate}[3][\TruncateMarker]{#3}
```

File 532 **lwarp-turnthepage.sty**

§ 641 Package turnthepage

turnthepage (*Pkg*) turnthepage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{turnthepage}[2011/03/24]

\newcommand{\turnthepage}{}
```

File 533 **lwarp-twoup.sty**

§ 642 Package twoup

twoup (*Pkg*) twoup is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{twoup}[2007/02/26]
```

```
\newcommand{\cleartolastpage}{}
```

File 534 **lwarp-txfonts.sty**

§ 643 Package

txfonts

(*Emulates or patches code by* Young Ryu.)

txfonts (*Pkg*) txfonts is used as-is for svg math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{txfonts}[2008/01/22]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{txfonts}

\LWR@mathjax@addgreek@l@up{}{up}
\end{warpMathJax}
```

File 535 **lwarp-txgreeks.sty**

§ 644 Package

txgreeks

(*Emulates or patches code by* Jean-François Burnol.)

txgreeks (*Pkg*) txgreeks is used as-is for svg math, and is emulated for MathJax.

The MathJax emulation honors all package options.

for HTML output:

```
\LWR@ProvidesPackagePass{txgreeks}[2011/03/16]

\LWR@infoprocessingmathjax{txgreeks}

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\iftgs@uplower% upright lowercase Greek
    \LWR@mathjax@addgreek@l@up{}{}
    \LWR@mathjax@addgreek@l@it{other}{}
\else% italic lowercase Greek
    \LWR@mathjax@addgreek@l@it{}{}
    \LWR@mathjax@addgreek@l@up{other}{}
\fi

\iftgs@itupper % italic uppercase Greek
    \LWR@mathjax@addgreek@u@it*{}{}
    \LWR@mathjax@addgreek@u@up*{other}{}
    \LWR@mathjax@addgreek@u@up*{var}{}
\else% upright uppercase Greek
    \LWR@mathjax@addgreek@u@up*{}{}
    \LWR@mathjax@addgreek@u@it*{other}{}
```

```
    \LWR@mathjax@addgreek@u@it*{var}{}
\fi
\end{warpMathJax}
```

File 536 **lwarp-typearea.sty**

§ 645 Package typearea

(*Emulates or patches code by* Markus Kohm.)

typearea (*Pkg*) typearea is emulated.

This package may be loaded standalone, but is also loaded automatically if koma-script classes are in use. \DeclareDocumentCommand is used to overwrite the koma-script definitions.

for HTML output:

```
\LWR@ProvidesPackageDrop{typearea}[2018/03/30]
```

```
\DeclareDocumentCommand{\typearea}{o m}{}
\DeclareDocumentCommand{\recalctypearea}{}{}
\@ifundefined{footheight}{\newlength\footheight}{}
\DeclareDocumentCommand{\areaset}{o m m}{}
\DeclareDocumentCommand{\activateareas}{}{}
\DeclareDocumentCommand{\storeareas}{m}{}
\DeclareDocumentCommand{\BeforeRestoreareas}{s m}{}
\DeclareDocumentCommand{\AfterRestoreareas}{s m}{}
\DeclareDocumentCommand{\AfterCalculatingTypearea}{s m}{}
\DeclareDocumentCommand{\AfterSettingArea}{s m}{}
```

File 537 **lwarp-typicons.sty**

§ 646 Package typicons

(*Emulates or patches code by* Arthur Vigil, Xavier Danaux.)

typicons (*Pkg*) typicons is patched for use by lwarp.

If \ticon is used, the name of the icon is used in the alt tag. Otherwise, for each of the individual icon macros, a generic alt tag is used.

for HTML output:

```
\LWR@ProvidesPackagePass{typicons}[2015/05/20]
```

```
\LetLtxMacro\LWR@orig@symbol\symbol

\let\LWR@orig@typicon@TI\TI

\newcommand*{\LWR@typicon@symbol}[1]{%
    \begin{lateximage}*[typicon][typicon#1]%
    \begingroup%
    \LWR@orig@typicon@TI%
    \LWR@orig@symbol{#1}%
    \endgroup%
    \end{lateximage}%
}
```

```
14
15 \renewcommand*{\TI}{%
16     \LetLtxMacro\symbol\LWR@typicon@symbol%
17 }
18
19 \renewcommand*{\ticon}[1]
20 {%
21     \begin{lateximage}*[#1 icon][typicon#1]%
22     \TI\csname ticon@#1\endcsname%
23     \end{lateximage}%
24 }
```

File 538 **lwarp-ulem.sty**

§ 647 Package ulem

(*Emulates or patches code by* Donald Arseneau.)

ulem (*Pkg*) Patched for use by lwarp.

for HTML output: Use the original package:

```
1 \LWR@ProvidesPackagePass{ulem}[2012/05/18]
```

Basic markup commands, using css:

```
2 \NewDocumentCommand{\LWR@HTML@uline}{+m}{%
3     \InlineClass%
4         (text-decoration:underline; text-decoration-skip: auto)%
5         {uline}{\LWR@isolate{#1}}%
6 }
7 \LWR@formatted{uline}
8
9 \NewDocumentCommand{\LWR@HTML@uuline}{+m}{%
10     \InlineClass%
11         (%
12             text-decoration:underline; text-decoration-skip: auto;%
13             text-decoration-style:double%
14         )%
15         {uuline}{\LWR@isolate{#1}}%
16 }
17 \LWR@formatted{uuline}
18
19 \NewDocumentCommand{\LWR@HTML@uwave}{+m}{%
20     \InlineClass%
21         (%
22             text-decoration:underline; text-decoration-skip: auto;%
23             text-decoration-style:wavy%
24         )%
25         {uwave}{\LWR@isolate{#1}}%
26 }
27 \LWR@formatted{uwave}
28
29 \NewDocumentCommand{\LWR@HTML@sout}{+m}{%
30     \InlineClass%
31         (text-decoration:line-through)%
32         {sout}{\LWR@isolate{#1}}%
```

```
}
\LWR@formatted{sout}

\NewDocumentCommand{\LWR@HTML@xout}{+m}{%
    \InlineClass%
        (text-decoration:line-through)%
        {xout}{\LWR@isolate{#1}}%
}
\LWR@formatted{xout}

\NewDocumentCommand{\LWR@HTML@dashuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline;%
            text-decoration-skip: auto;%
            text-decoration-style:dashed%
        )%
        {dashuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{dashuline}

\NewDocumentCommand{\LWR@HTML@dotuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline;%
            text-decoration-skip: auto;%
            text-decoration-style: dotted%
        )%
        {dotuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{dotuline}
```

Nullified/emulated macros:

```
\NewDocumentCommand{\LWR@HTML@markoverwith}{m}{}
\LWR@formatted{markoverwith}

\NewDocumentCommand{\LWR@HTML@ULon}{+m}{\uline{#1}\egroup}
\LWR@formatted{ULon}
```

File 539 **lwarp-umoline.sty**

§ 648 Package umoline

(*Emulates or patches code by* Hiroshi Nakashima.)

umoline (*Pkg*) umoline is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{umoline}[2000/07/11]

\newcommand*{\LWR@HTML@Underline}[1]{%
    \InlineClass{uline}{#1}%
}
\LWR@formatted{Underline}

\newcommand*{\LWR@HTML@Midline}[1]{%
    \InlineClass{sout}{#1}%
```

```
}
\LWR@formatted{Midline}

\newcommand*{\LWR@HTML@Overline}[1]{%
    \InlineClass{oline}{#1}%
}
\LWR@formatted{Overline}

\newcommand*{\LWR@HTML@UMOline}[2]{%
    \InlineClass{uline}{#2}%
}
\LWR@formatted{UMOline}

\NewDocumentCommand{\LWR@HTML@UMOspace}{s m o}{\hspace*{#2}}
\LWR@formatted{UMOspace}

\NewDocumentCommand{\LWR@HTML@UMOnewline}{s}{\newline}
\LWR@formatted{UMOnewline}
```

File 540 **lwarp-underscore.sty**

§ 649 Package

underscore

underscore (*Pkg*) underscore is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{underscore}[2006/09/13]
```

File 541 **lwarp-unicode-math.sty**

§ 650 Package

unicode-math

(*Emulates or patches code by* Will Robertson.)

unicode-math (*Pkg*) unicode-math is supported as-is for HTML with SVGmath.

⚠ MathJax If the document source includes embedded Unicode characters, these may not be reproduced correctly for *pdftotext*, and thus not display correctly in MathJax.

Symbol font commands are emulated, but not all combinations are supported by MathJax, especially with the dedicated Greek macros. Symbol macros such as `\symbfsf` may not be sans or bold. For Greek, use the Unicode equivalent, if necessary.

⚠ `\mathversion` The MathJax emulation does not change with the use of `\mathversion`. Whatever emulation is established at the begin of the document will remain.

The option `sans-style` honors `upright` and `italic`, but `italic` will not be sans, in order to support Greek macros.

Greek macros such as `\alpha` respond to the `math-style` option. Latin symbols does not, per MathJax limitations, unless placed inside `\symbit` or similar.

Macros from the categories `\mathopen`, `\mathclose`, and `\mathfence` are emulated. Due to current MathJax limitations, not all stretch to the correct height.

Also emulated are macros from the categories \mathpunct, \mathover, \mathunder, \mathaccent, \mathbotaccent, and \mathop.

The individual unicode-math macros of categories \mathbin, \mathord, and \mathrel are not emulated for MathJax, as there are more than two thousand of them, but they may be added as needed. Place the following in the document preamble after loading unicode-math, including a definition for each macro which is used in the document but undefined in MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\uplus}{\mathbin{\unicode{x0228E}}}}
...
\end{warpMathJax}
```

Use \mathrel, \mathbin, etc. depending on the category of each macro. For a list of macro names and symbols, see **texdoc unimath-symbols**.

for HTML output:

```
\LWR@ProvidesPackagePass{unicode-math}[2019/09/26]

\LWR@origRequirePackage{lwarp-common-mathjax-letters}

\begin{warpMathJax}
\LWR@infoprocessingmathjax{unicode-math}

% Not all are possible in MathJax.
\CustomizeMathJax{\let\symnormal\mathit}
\CustomizeMathJax{\let\symliteral\mathrm}
\CustomizeMathJax{\let\symbb\mathbb}
\CustomizeMathJax{\let\symbbit\mathbb}% not italic
\CustomizeMathJax{\let\symcal\mathcal}
\CustomizeMathJax{\let\symscr\mathscr}
\CustomizeMathJax{\let\symfrak\mathfrak}

\CustomizeMathJax{\let\symsfup\mathsf}

\CustomizeMathJax{\let\symsfit\mathit}% not sans
% \CustomizeMathJax{\newcommand{\symsfit}[1]{%
%     \mmlToken{mi}[mathvariant="sans-serif-italic"]{#1}}% not greek
% }

\CustomizeMathJax{\let\symbfsf\mathbf}% not sans
% \CustomizeMathJax{\newcommand{\symbfsf}[1]{%
%     \mmlToken{mi}[mathvariant="bold-sans-serif"]{#1}}% not greek
% }

\CustomizeMathJax{\let\symbfup\mathbf}
\CustomizeMathJax{\newcommand{\symbfit}[1]{\boldsymbol{#1}}}
\CustomizeMathJax{\let\symbfcal\mathcal}% not bold

\CustomizeMathJax{\let\symbfscr\mathscr}% not bold
% \CustomizeMathJax{\newcommand{\symbfscr}[1]{
%     \mmlToken{mi}[mathvariant="math-bold-script"]{#1}}% not greek
% }

\CustomizeMathJax{\let\symbffrak\mathfrak}% not bold
% \CustomizeMathJax{\newcommand{\symbffrak}[1]{%
%     \mmlToken{mi}[mathvariant="math-bold-fraktur"]{#1}}% not greek
% }
```

```
\CustomizeMathJax{\let\symbfsfup\mathbf}% not sans
% \CustomizeMathJax{\newcommand{\symbfsfup}[1]{%
%     \mmlToken{mi}[mathvariant="bold-sans-serif"]{#1}}% not greek
% }

\CustomizeMathJax{\newcommand{\symbfsfit}[1]{\boldsymbol{#1}}}% not sans
% \CustomizeMathJax{\newcommand{\symbfsfit}[1]{%
%     \mmlToken{mi}[mathvariant="sans-serif-bold-italic"]{#1}}% not greek
% }

% Duplicates below are commented out.
\CustomizeMathJax{\let\symup\mathrm}
\CustomizeMathJax{\let\symbf\mathbf}% \symbfup defined above
\CustomizeMathJax{\let\symit\mathit}
% \CustomizeMathJax{\let\symbfit\mathit}% not bold


\ExplSyntaxOn
\AtBeginDocument{
\bool_if:NTF \g__um_sfliteral_bool
    {\CustomizeMathJax{\let\symsf\symsfup}}
    {
        \bool_if:NTF \g__um_upsans_bool
            {\CustomizeMathJax{\let\symsf\symsfup}}
            {\CustomizeMathJax{\let\symsf\symsfit}}
    }
}
\ExplSyntaxOff

% \CustomizeMathJax{\let\symbfsfup\mathbf}% not sans
% \CustomizeMathJax{\let\symsfit\mathit}% not sans
% \CustomizeMathJax{\let\symbfsfit\mathit}% not bold nor sans
\CustomizeMathJax{\let\symtt\mathtt}
% \CustomizeMathJax{\let\symbb\mathbb}
% \CustomizeMathJax{\let\symbbit\mathbb}% not italic
% \CustomizeMathJax{\let\symscr\mathscr}
% \CustomizeMathJax{\let\symbfscr\mathscr}% not bold
% \CustomizeMathJax{\let\symfrak\mathfrak}
\CustomizeMathJax{\let\symbffrac\mathbffrac}
```

Some symbol categories defined by unicode-math, in case they are used inside custom macros:

```
\CustomizeMathJax{\newcommand{\mathfence}[1]{\mathord{#1}}}
\CustomizeMathJax{\newcommand{\mathover}[1]{#1}}
\CustomizeMathJax{\newcommand{\mathunder}[1]{#1}}
\CustomizeMathJax{\newcommand{\mathaccent}[1]{#1}}
\CustomizeMathJax{\newcommand{\mathbotaccent}[1]{#1}}
\CustomizeMathJax{\newcommand{\mathalpha}[1]{\mathord{#1}}}
```

`math-style` is one of: `ISO`, `TeX`, `french`, `upright`, or `literal`, which set `\g__um_upGreek_bool` and `\g__um_upgreek_bool`.

```
\ExplSyntaxOn

\AtBeginDocument{
\bool_if:NTF \g__um_upGreek_bool
    {\LWR@mathjax@addgreek@u@up*{}{}}
```

```
    {\LWR@mathjax@addgreek@u@it*{}{}}

\bool_if:NTF \g__um_upgreek_bool
    {\LWR@mathjax@addgreek@l@up{}{}}
    {\LWR@mathjax@addgreek@l@it{}{}}
}

\LWR@mathjax@addgreek@u@up*{up}{}
\LWR@mathjax@addgreek@u@it*{it}{}
\LWR@mathjax@addgreek@l@up{up}{}
\LWR@mathjax@addgreek@l@it{it}{}

\ExplSyntaxOff

\CustomizeMathJax{\let\lparen(}
\CustomizeMathJax{\let\rparen)}
\CustomizeMathJax{\newcommand{\cuberoot}[1]{\,{}^3\!\!\sqrt{#1}}\,}
\CustomizeMathJax{\newcommand{\fourthroot}[1]{\,{}^4\!\!\sqrt{#1}}\,}
```

Many `\mathopen` / `\mathclose` delimiters are defined in `lwarp_mathjax.txt`, where `\left` / `\right` support is added.

```
\CustomizeMathJax{\newcommand{\longdivision}[1]{\mathord{\unicode{x027CC}#1}}}

\CustomizeMathJax{\newcommand{\mathcomma}{,}}
\CustomizeMathJax{\newcommand{\mathcolon}{:}}
\CustomizeMathJax{\newcommand{\mathsemicolon}{;}}

\CustomizeMathJax{\newcommand{\overbracket}[1]{\mathinner{\overline{\ulcorner{#1}\urcorner}}}}
\CustomizeMathJax{\newcommand{\underbracket}[1]{\mathinner{\underline{\llcorner{#1}\lrcorner}}}}

\CustomizeMathJax{\newcommand{\overbar}[1]{\mathord{#1\unicode{x00305}}}}
\CustomizeMathJax{\newcommand{\ovhook}[1]{\mathord{#1\unicode{x00309}}}}
\CustomizeMathJax{\newcommand{\ocirc}[1]{\mathord{#1\unicode{x0030A}}}}
\CustomizeMathJax{\newcommand{\candra}[1]{\mathord{#1\unicode{x00310}}}}
\CustomizeMathJax{\newcommand{\oturnedcomma}[1]{\mathord{#1\unicode{x00312}}}}
\CustomizeMathJax{\newcommand{\ocommatopright}[1]{\mathord{#1\unicode{x00315}}}}
\CustomizeMathJax{\newcommand{\droang}[1]{\mathord{#1\unicode{x0031A}}}}
\CustomizeMathJax{\newcommand{\leftharpoonaccent}[1]{\mathord{#1\unicode{x020D0}}}}
\CustomizeMathJax{\newcommand{\rightharpoonaccent}[1]{\mathord{#1\unicode{x020D1}}}}
\CustomizeMathJax{\newcommand{\vertoverlay}[1]{\mathord{#1\unicode{x020D2}}}}
\CustomizeMathJax{\newcommand{\leftarrowaccent}[1]{\mathord{#1\unicode{x020D0}}}}
\CustomizeMathJax{\newcommand{\annuity}[1]{\mathord{#1\unicode{x020E7}}}}
\CustomizeMathJax{\newcommand{\widebridgeabove}[1]{\mathord{#1\unicode{x020E9}}}}
\CustomizeMathJax{\newcommand{\asteraccent}[1]{\mathord{#1\unicode{x020F0}}}}
\CustomizeMathJax{\newcommand{\threeunderdot}[1]{\mathord{#1\unicode{x020E8}}}}

\CustomizeMathJax{\newcommand{\Bbbsum}{\mathop{\unicode{x2140}}\limits}}
\CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
\CustomizeMathJax{\newcommand{\intclockwise}{\mathop{\unicode{x2231}}\limits}}
\CustomizeMathJax{\newcommand{\ointclockwise}{\mathop{\unicode{x2232}}\limits}}
\CustomizeMathJax{\newcommand{\ointctrclockwise}{\mathop{\unicode{x2233}}\limits}}
\CustomizeMathJax{\newcommand{\varointclockwise}{\mathop{\unicode{x2232}}\limits}}
\CustomizeMathJax{\newcommand{\leftouterjoin}{\mathop{\unicode{x27D5}}\limits}}
\CustomizeMathJax{\newcommand{\rightouterjoin}{\mathop{\unicode{x27D6}}\limits}}
\CustomizeMathJax{\newcommand{\fullouterjoin}{\mathop{\unicode{x27D7}}\limits}}
\CustomizeMathJax{\newcommand{\bigbot}{\mathop{\unicode{x27D8}}\limits}}
\CustomizeMathJax{\newcommand{\bigtop}{\mathop{\unicode{x27D9}}\limits}}
\CustomizeMathJax{\newcommand{\xsol}{\mathop{\unicode{x29F8}}\limits}}
```

```
\CustomizeMathJax{\newcommand{\xbsol}{\mathop{\unicode{x29F9}}\limits}}
\CustomizeMathJax{\newcommand{\bigcupdot}{\mathop{\unicode{x2A03}}\limits}}
\CustomizeMathJax{\newcommand{\bigsqcap}{\mathop{\unicode{x2A05}}\limits}}
\CustomizeMathJax{\newcommand{\conjquant}{\mathop{\unicode{x2A07}}\limits}}
\CustomizeMathJax{\newcommand{\disjquant}{\mathop{\unicode{x2A08}}\limits}}
\CustomizeMathJax{\newcommand{\bigtimes}{\mathop{\unicode{x2A09}}\limits}}
\CustomizeMathJax{\newcommand{\modtwosum}{\mathop{\unicode{x2A0A}}\limits}}
\CustomizeMathJax{\newcommand{\sumint}{\mathop{\unicode{x2A0B}}\limits}}
\CustomizeMathJax{\newcommand{\intbar}{\mathop{\unicode{x2A0D}}\limits}}
\CustomizeMathJax{\newcommand{\intBar}{\mathop{\unicode{x2A0E}}\limits}}
\CustomizeMathJax{\newcommand{\fint}{\mathop{\unicode{x2A0F}}\limits}}
\CustomizeMathJax{\newcommand{\cirfnint}{\mathop{\unicode{x2A10}}\limits}}
\CustomizeMathJax{\newcommand{\awint}{\mathop{\unicode{x2A11}}\limits}}
\CustomizeMathJax{\newcommand{\rppolint}{\mathop{\unicode{x2A12}}\limits}}
\CustomizeMathJax{\newcommand{\scpolint}{\mathop{\unicode{x2A13}}\limits}}
\CustomizeMathJax{\newcommand{\npolint}{\mathop{\unicode{x2A14}}\limits}}
\CustomizeMathJax{\newcommand{\pointint}{\mathop{\unicode{x2A15}}\limits}}
\CustomizeMathJax{\newcommand{\sqint}{\mathop{\unicode{x2A16}}\limits}}
\CustomizeMathJax{\newcommand{\intlarhk}{\mathop{\unicode{x2A17}}\limits}}
\CustomizeMathJax{\newcommand{\intx}{\mathop{\unicode{x2A18}}\limits}}
\CustomizeMathJax{\newcommand{\intcap}{\mathop{\unicode{x2A19}}\limits}}
\CustomizeMathJax{\newcommand{\intcup}{\mathop{\unicode{x2A1A}}\limits}}
\CustomizeMathJax{\newcommand{\upint}{\mathop{\unicode{x2A1B}}\limits}}
\CustomizeMathJax{\newcommand{\lowint}{\mathop{\unicode{x2A1C}}\limits}}
\CustomizeMathJax{\newcommand{\bigtriangleleft}{\mathop{\unicode{x2A1E}}\limits}}
\CustomizeMathJax{\newcommand{\zcmp}{\mathop{\unicode{x2A1F}}\limits}}
\CustomizeMathJax{\newcommand{\zpipe}{\mathop{\unicode{x2A20}}\limits}}
\CustomizeMathJax{\newcommand{\zproject}{\mathop{\unicode{x2A21}}\limits}}
\CustomizeMathJax{\newcommand{\biginterleave}{\mathop{\unicode{x2AFC}}\limits}}
\CustomizeMathJax{\newcommand{\bigtalloblong}{\mathop{\unicode{x2AFF}}\limits}}
\CustomizeMathJax{\newcommand{\arabicmaj}{\mathop{\unicode{x1EEF0}}\limits}}
\CustomizeMathJax{\newcommand{\arabichad}{\mathop{\unicode{x1EEF1}}\limits}}

\end{warpMathJax}
```

File 542 **lwarp-units.sty**

§ 651 Package units

(*Emulates or patches code by* Axel Reichert.)

`units` (*Pkg*) units is patched for use by lwarp.

Values are not styled by CSS, and take the style of the surrounding HTML text.

Units are styled according to the print version, so they will be forced to upright roman in HTML if the print version does so. It may be necessary to adjust the document's body CSS to match the print version.

for HTML output:

```
\LWR@ProvidesPackagePass{units}[1998/08/04]

\DeclareRobustCommand*{\LWR@HTML@unit}[2][]{%
\ifblank{#1}%
    {\LWR@textcurrentfont{#2}}%
    {%
        #1%
        \ifthenelse{\boolean{B@UnitsLoose}}{~}{\,}%
```

```
        \LWR@textcurrentfont{#2}%
    }%
}
\LWR@formatted{unit}

\DeclareRobustCommand*{\LWR@HTML@unitfrac}[3][]{%
\ifblank{#1}%
    {%
            \nicefrac{#2}{#3}%
    }%
    {%
            #1%
            \ifthenelse{\boolean{B@UnitsLoose}}{~}{\,}%
            \nicefrac{#2}{#3}%
    }%
}

\LWR@formatted{unitfrac}
```

For MATHJAX:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\unit}[2][]{#1 \mathinner{#2}}}
\CustomizeMathJax{\newcommand{\unitfrac}[3][]{#1 \mathinner{{}^{#2}\!/\!_{#3}}}}
\end{warpMathJax}
```

File 543 **lwarp-unitsdef.sty**

§ 652 Package

unitsdef

(*Emulates or patches code by* PATRICK HAPPEL.)

unitsdef (*Pkg*) unitsdef is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{unitsdef}[2005/01/04]

\newcommand{\LWR@HTML@unitvaluesep}{\,}
\LWR@formatted{unitvaluesep}

\newcommand{\LWR@HTML@unittimes}{\@@setunitsepfalse\HTMLunicode{22c5}}% \cdot
\LWR@formatted{unittimes}

\newunit{\LWR@HTML@arcmin}{%
    \HTMLunicode{2032}% prime
}
\LWR@formatted{arcmin}

\newunit{\LWR@HTML@arcsec}{%
    \HTMLunicode{2033}% dbl prime
}
\LWR@formatted{arcsec}

\newrobustcmd{\LWR@HTML@SI}[2]{%
  \begingroup%
    \let\unit@@xspace\relax%
    \unitSIdef\selectfont%
      \LWR@textcurrentfont{#1#2}% lwarp
```

```
  \endgroup%
}
\LWR@formatted{SI}
```

File 544 **lwarp-upgreek.sty**

§ 653 Package upgreek

(*Emulates or patches code by* Walter Schmidt.)

upgreek (*Pkg*) upgreek is used as-is for svg math, and is emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{upgreek}[2003/02/12]
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\require{upgreek}}
\end{warpMathJax}
```

File 545 **lwarp-upref.sty**

§ 654 Package upref

upref (*Pkg*) upref is ignored.

for HTML output: Discard all options for lwarp-upref:

```
\LWR@ProvidesPackageDrop{upref}[2007/03/14]
```

File 546 **lwarp-url.sty**

§ 655 Package url

(*Emulates or patches code by* Donald Arseneau.)

url (*Pkg*) url is patched for use by lwarp.

for HTML output:

```
\LetLtxMacro\LWR@url@orig@url\LWR@url

\LWR@ProvidesPackagePass{url}[2013/09/16]

\newcommand*{\LWR@HTML@Url@FormatString}{%
    \expandafter\LWR@url@orig@url\expandafter{\Url@String}%
}
\LWR@formatted{Url@FormatString}
```

File 547 **lwarp-ushort.sty**

§ 656 Package ushort

(*Emulates or patches code by* Martin Väth.)

ushort (*Pkg*) ushort is used as-is, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{ushort}[2001/06/13]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\ushortdline}[1]{%
    \kern{.1em}\underline{\underline{{#1}}}\kern{.1em}%
}}
\CustomizeMathJax{\newcommand{\ushort}[1]{\kern{.1em}\underline{#1}\kern{.1em}}}
\CustomizeMathJax{\newcommand{\ushortd}[1]{\ushortdline{#1}}}
\CustomizeMathJax{\newcommand{\ushortw}[1]{\kern{.1em}\underline{#1}\kern{.1em}}}
\CustomizeMathJax{\newcommand{\ushortdw}[1]{\ushortdline{#1}}}
\end{warpMathJax}
```

File 548 **lwarp-uspace.sty**

§ 657 Package uspace

uspace (*Pkg*) uspace is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{uspace}[2016/11/06]
```

File 549 **lwarp-varioref.sty**

§ 658 Package varioref

(*Emulates or patches code by* Frank Mittelbach.)

varioref (*Pkg*) varioref is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{varioref}[2020/01/23]
```

Page-related output is not used for HTML output.

```
\def\reftextfaceafter {\unskip}%
\def\reftextfacebefore{\unskip}%
\def\reftextafter     {\unskip}%
\def\reftextbefore    {\unskip}%
\def\reftextcurrent   {\unskip}%
\def\reftextfaraway#1{\unskip}%
\def\reftextpagerange#1#2{\unskip}%
```

File 550 **lwarp-verse.sty**

§ 659 Package

verse

(*Emulates or patches code by* Peter Wilson.)

verse (*Pkg*) verse is supported and patched by lwarp.

for HTML output: Pass all options for lwarp-verse:

```
\LWR@ProvidesPackagePass{verse}[2009/09/04]
```

When using verse or memoir, always place a \\ after each line.

\attrib The documentation for the verse and memoir packages suggest defining an \attrib command, which may already exist in current documents, but it will only work for print output. lwarp provides \attribution, which works for both print and HTML output. To combine the two so that \attrib is used for print and \attribution is used for HTML:

```
\begin{warpHTML}
\let\attrib\attribution
\end{warpHTML}
```

\vleftskip (*Len*)
\vleftmargini (*Len*)
\HTMLvleftskip (*Len*)
\HTMLleftmargini (*Len*)

These lengths are used by verse and memoir to control the left margin, and they may already be set by the user for print output. New lengths \HTMLvleftskip and \HTMLleftmargini are provided to control the margins in HTML output. These new lengths may be set by the user before any verse environment, and persist until they are manually changed again. One reason to change \HTMLleftmargini is if there is a wide \flagverse in use, such as the word "Chorus", in which case the value of \HTMLleftmargini should be set to a wide enough length to contain "Chorus". The default is wide enough for a stanza number.

⚠ spacing

⚠ verse margin

Horizontal spacing relies on *pdftotext*'s ability to discern the layout (-layout option) of the text in the HTML-tagged PDF output. For some settings of \HTMLleftmargini or \HTMLleftskip the horizontal alignment may not work out exactly, in which case a label may be shifted by one space. During translation to HTML, the stanza numbers are kept out of the left margin, which would have caused *pdftotext* to shift everything over.

verse (*env.*) The verse environment will be placed inside a HTML <pre>.

```
\AfterEndPreamble{
\LWR@traceinfo{Patching verse.}
```

At the beginning of the verse environment:

```
\AtBeginEnvironment{verse}
{%
```

Use the original list environment inside a <pre> to attempt to preserve formatting.

```
\LWR@restoreoriglists%
```

verse (*Pkg*)
memoir (*Cls*)
\flagverse
\vleftskip (*Len*)

The verse or memoir packages can place stanza numbers to the left with their \flagverse command. The following does not allow them to go into the left margin, which would cause *pdfcrop* to crop the entire page further to the left.

```
\ifdef{\vleftskip}{%
\setlength{\vleftskip}{\HTMLvleftskip}
\setlength{\leftmargini}{\HTMLleftmargini}
}{}
\LWR@forcenewpage
\LWR@atbeginverbatim{verse}%
}
```

After the end of the verse environment, which places the <pre> tag at the regular left margin:

```
\AtEndEnvironment{verse}{%
\leavevmode%
\LWR@afterendverbatim%
}
```

Patch to place poemtitle inside an HTML of class poemtitle:

```
\ifdef{\poemtitle}{
\DeclareDocumentCommand{\@vstypeptitle}{m}{%
   \vspace{\beforepoemtitleskip}%
   {\InlineClass{poemtitle}{\poemtitlefont #1}\par}%
   \vspace{\afterpoemtitleskip}%
 }
}{}

\LWR@traceinfo{Finished patching verse.}
}% AfterEndPreamble
```

File 551 **lwarp-versonotes.sty**

§ 660 Package

versonotes

(*Emulates or patches code by* Norman Gray.)

versonotes (*Pkg*)

versonotes is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{versonotes}[2019/07/06]

\newcommand{\versonote}[1]{\marginpar{#1}}
\newdimen\versotextwidth
\newdimen\versoleftmargin
\newcommand*{\versolayout}{}
```

In case the user changed the page number before loading versonotes:

```
\setcounter{page}{1}
```

File 552 **lwarp-vertbars.sty**

§ 661 Package vertbars

(*Emulates or patches code by* Peter Wilson.)

vertbars (*Pkg*) vertbars is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{vertbars}[2010/11/27]

\newlength{\barwidth}
\setlength{\barwidth}{0.4pt}
\newlength{\barspace}
\setlength{\barspace}{1em}

\newenvironment{vertbar}{
    \LWR@forcenewpage
    \LWR@forceminwidth{\barwidth}
    \begin{BlockClass}[%
        border-left: \LWR@printlength{\LWR@atleastonept} solid black ; %
        padding-left: \LWR@printlength{\barspace}%
    ]{vertbar}
}{
    \end{BlockClass}
}
```

File 553 **lwarp-vmargin.sty**

§ 662 Package vmargin

vmargin (*Pkg*) vmargin is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{vmargin}[2004/07/15]

\newcommand*{\LWRVM@customsize}[2]{}
\newcommand*{\setpapersize}[2][]{\ifstrequal{#2}{custom}{\LWRVM@customsize}{}}
\newcommand*{\setmargins}[8]{}
\newcommand*{\setmarginsrb}[8]{}
\newcommand*{\setmargnohf}[4]{}
\newcommand*{\setmargnohfrb}[4]{}
\newcommand*{\setmarg}[4]{}
\newcommand*{\setmargrb}[4]{}
\newlength{\PaperWidth}
\setlength{\PaperWidth}{8.5in}
\newlength{\PaperHeight}
\setlength{\PaperHeight}{11in}
\newif\ifLandscape
```

File 554 **lwarp-vowel.sty**

§ 663 Package vowel

(*Emulates or patches code by* FUKUI REI.)

vowel (*Pkg*) vowel is patched for use by lwarp.

This package has been tested with *pdflatex* and the Type 1 TIPA fonts using the following package load sequence:

```
\usepackage[T3,T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[noenc]{tipa}
\usepackage{vowel}
```

for HTML output:

```
\LWR@ProvidesPackagePass{vowel}[2002/08/08]

\renewenvironment{vowel}[1][]
    {%
        \begin{lateximage}[-vowel-~\PackageDiagramAltText]%
        \@vowel[#1]%
    }
    {%
        \@@vowel%
        \end{lateximage}%
    }
```

File 555 **lwarp-vpe.sty**

§ 664 Package vpe

vpe (*Pkg*) vpe is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{vpe}[2012/04/18]
```

File 556 **lwarp-vwcol.sty**

§ 665 Package vwcol

(*Emulates or patches code by* WILL ROBERTSON.)

vwcol (*Pkg*) vwcol is patched for use with lwarp.

The `width` option is ignored. All `vwcol` environments adjust to 1–3 equal-width columns, depening on the width of the browser window.

The remaining options are supported, except for `lines` and `maxrecursion`.

for HTML output:

```
\LWR@ProvidesPackagePass{vwcol}[2015/02/10]
```

Factored from \vwcol. Each is given a style tag to append to the final style.

\LWR@vwcol@addrule {⟨*style tag*⟩}

```
\newcommand*{\LWR@vwcol@addrule}[1]{%
    \appto{\LWR@vwcolstyle}{%
        #1: %
     \LWR@printlength{\vwcol@rule} solid \LWR@origpound\LWR@vwcol@rulecolor ; %
    }%
}
```

\LWR@vwcol@addrule {⟨*style tag*⟩}

```
\newcommand*{\LWR@vwcol@addgap}[1]{%
    \appto{\LWR@vwcolstyle}{%
        #1: %
        \LWR@printlength{\vwcol@sep} ; %
    }%
}
```

Env vwcol {⟨*key/values*⟩}

Redefine the environment to add a HTML style. The style is built depending on the required options.

```
\renewenvironment*{vwcol}[1][]{%
```

New paragraph, and process the options:

```
\LWR@stoppars%
\vwcolsetup{#1}%
```

Begin with no style:

```
\newcommand*{\LWR@vwcolstyle}{}
```

presep and postsep are created with HTML margins:

```
\if@vwcol@presep
    \appto{\LWR@vwcolstyle}{margin-left: 1em ; padding-left: .5em ; }
\fi
\if@vwcol@postsep
    \appto{\LWR@vwcolstyle}{margin-right: 1em ; padding-right: .5em ; }
\fi
```

sep becomes column-gap:

```
\ifdimgreater{\vwcol@sep}{1sp}{
    \LWR@vwcol@addgap{column-gap}
    \LWR@vwcol@addgap{-moz-column-gap}
    \LWR@vwcol@addgap{-webkit-column-gap}
}{}
```

rule become column-rule, while prerule and postrule become HTML borders:

```
\convertcolorspec{named}{\vwcol@rulecol}{HTML}\LWR@vwcol@rulecolor%
\ifdimgreater{\vwcol@rule}{0pt}{
    \ifdimless{\vwcol@rule}{1pt}{
        \setlength{\vwcol@rule}{1pt}
    }{}
    \LWR@vwcol@addrule{column-rule}
    \LWR@vwcol@addrule{-moz-column-rule}
    \LWR@vwcol@addrule{-webkit-column-rule}
    \if@vwcol@prerule\LWR@vwcol@addrule{border-left}\fi
```

```
    \if@vwcol@postrule\LWR@vwcol@addrule{border-right}\fi
}{}
```

Each of the justify options becomes a `text-align`. Indentation is added where appropriate.

```
\ifdefequal{\vwcol@justify}{\RaggedRight}{
    \appto{\LWR@vwcolstyle}{text-align: left ; }
    \ifdimgreater{\vwcol@parindent}{0pt}{
        \appto{\LWR@vwcolstyle}{%
            text-indent: \LWR@printlength{\vwcol@parindent} ; %
        }
    }{}
}{}

\ifdefequal{\vwcol@justify}{\RaggedLeft}{
    \appto{\LWR@vwcolstyle}{text-align: right ; }
}{}

\ifdefequal{\vwcol@justify}{\Centering}{
    \appto{\LWR@vwcolstyle}{text-align: center ; }
}{}

\ifdefequal{\vwcol@justify}{\justifying}{
    \appto{\LWR@vwcolstyle}{text-align: justify ; }
    \ifdimgreater{\vwcol@parindent}{0pt}{
        \appto{\LWR@vwcolstyle}{%
            text-indent: \LWR@printlength{\vwcol@parindent} ; %
        }
    }{}
}{}
```

Create the with the assembled style:

```
\BlockClass[\LWR@vwcolstyle]{multicols}
}
```

When the environment ends:

```
{
    \endBlockClass
    \LWR@startpars
}
```

File 557 **lwarp-wallpaper.sty**

§ 666 Package wallpaper

(*Emulates or patches code by* Michael H.F. Wilkinson.)

`wallpaper` (*Pkg*) wallpaper is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{wallpaper}[2005/01/18]

\newcommand*{\CenterWallPaper}[2]{}
\newcommand*{\ThisCenterWallPaper}[2]{}
\newcommand*{\TileWallPaper}[3]{}
\newcommand*{\ThisTileWallPaper}[3]{}
\newcommand*{\TileSquareWallPaper}[2]{}
\newcommand*{\ThisTileSquareWallPaper}[2]{}
\newcommand*{\ULCornerWallPaper}[2]{}
```

```
\newcommand*{\ThisULCornerWallPaper}[2]{}
\newcommand*{\LLCornerWallPaper}[2]{}
\newcommand*{\ThisLLCornerWallPaper}[2]{}
\newcommand*{\URCornerWallPaper}[2]{}
\newcommand*{\ThisURCornerWallPaper}[2]{}
\newcommand*{\LRCornerWallPaper}[2]{}
\newcommand*{\ThisLRCornerWallPaper}[2]{}
\newcommand*{\ClearWallPaper}{}
\newlength{\wpXoffset}
\newlength{\wpYoffset}
```

File 558 **lwarp-watermark.sty**

§ 667 Package watermark

(*Emulates or patches code by* Alexander I. Rozhenko.)

watermark (*Pkg*) watermark is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{watermark}[2004/12/09]
```

```
\newcommand{\watermark}[1]{}
\newcommand{\leftwatermark}[1]{}
\newcommand{\rightwatermark}[1]{}
\newcommand{\thiswatermark}[1]{}
\newcommand{\thispageheading}[1]{}
```

File 559 **lwarp-widetable.sty**

§ 668 Package widetable

(*Emulates or patches code by* Claudio Beccari.)

widetable (*Pkg*) widetable is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{widetable}[2019-06-25]
```

```
\newenvironment{widetable}{\begin{tabular*}}{\end{tabular*}}
```

File 560 **lwarp-widows-and-orphans.sty**

§ 669 Package widows-and-orphans

widows-and-orphans (*Pkg*) widows-and-orphans is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{widows-and-orphans}[2018/09/01]
```

```
\NewDocumentCommand\WaOsetup{m}{}
\NewDocumentCommand\WaOparameters{}{}
\NewDocumentCommand\WaOignorenext{}{}
```

File 561 **lwarp-witharrows.sty**

§ 670 Package

witharrows

(*Emulates or patches code by* F. Pantigny.)

witharrows (*Pkg*) witharrows is patched for use by lwarp. Emulation is provided for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{witharrows}[2019/12/27]

\ifbool{mathjax}{
    % For the hidden print version in the HTML:
    \newcommand{\Arrow}[2][]{}
    \newcommand{\unicode}[1]{}
    \NewDocumentEnvironment { DispWithArrows } { ! d < > ! O { } +b}
        {
            \IfValueTF{#1}{
                \begin{displaymath}
                #1 \left\lbrace
                \begin{align}
                #3
                \end{align}
                \right .
                \end{displaymath}
            }{
                \begin{displaymath}
                \begin{align}
                #3
                \end{align}
                \end{displaymath}
            }
        }
        {}
    \NewDocumentEnvironment { DispWithArrows* } { ! d < > ! O { } +b}
        {
            \IfValueTF{#1}{
                \begin{displaymath}
                #1 \left\lbrace
                \begin{align*}
                #3
                \end{align*}
                \right .
                \end{displaymath}
            }{
                \begin{displaymath}
                \begin{align*}
                #3
                \end{align*}
                \end{displaymath}
            }
        }
        {}
}{
    % If not MathJax, use SVG images.
   \BeforeBeginEnvironment{WithArrows}{\global\booltrue{LWR@unknownmathsize}}
    \BeforeBeginEnvironment{DispWithArrows}{%
```

```
        \begin{BlockClass}{displaymathnumbered}%
        \begin{lateximage}%
    }
    \AfterEndEnvironment{DispWithArrows}{\end{lateximage}\end{BlockClass}}
    \BeforeBeginEnvironment{DispWithArrows*}{%
        \begin{BlockClass}{displaymath}%
        \begin{lateximage}%
    }
    \AfterEndEnvironment{DispWithArrows*}{\end{lateximage}\end{BlockClass}}
}

\begin{warpMathJax}
\CustomizeMathJax{\newenvironment{WithArrows}[1][]{\begin{aligned}}{\end{aligned}}}
% Unable to make a sized box.
\CustomizeMathJax{\newcommand{\Arrow}[2][]{&{\Large\unicode{x2938}}~\textit{#2}}}
\end{warpMathJax}
```

File 562 **lwarp-wrapfig.sty**

§ 671 Package wrapfig

(*Emulates or patches code by* Donald Arseneau.)

`wrapfig` (*Pkg*) wrapfig is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{wrapfig}[2003/01/31]

\newcommand*{\LWR@wrapposition}{}

\newcommand{\LWR@wrapfig@printHTMLwidth}{\LWR@printlength{\LWR@templengthone}}

\AtBeginDocument{
    \IfPackageLoadedTF{keyfloat}{
        \renewcommand{\LWR@wrapfig@printHTMLwidth}{%
            \ifboolexpr{
                test {\ifnumgreater{\value{KFLT@keyfloatdepth}}{0}} or
                bool {KFLT@inkeysubfloats}
            }%
              {\LWR@printpercentlength{\LWR@templengthone}{\linewidth}\%; }%
                 {\LWR@printlength{\LWR@templengthone}}%
        }%
    }{}
}

\newcommand*{\LWR@subwrapfigure}[2]{%
    \renewcommand*{\LWR@wrapposition}{}%
    \ifthenelse{%
        \equal{#1}{r}\OR\equal{#1}{R}\OR%
        \equal{#1}{o}\OR\equal{#1}{O}%
    }%
        {\renewcommand*{\LWR@wrapposition}{float:right}}%
        {\renewcommand*{\LWR@wrapposition}{float:left}}%
    \setlength{\LWR@templengthone}{#2}%
    \LWR@BlockClassWP{%
        width:\LWR@printlength{\LWR@templengthone}; \LWR@wrapposition; %
        margin:10pt%
    }%
```

```
    {%
        width:\LWR@wrapfig@printHTMLwidth; %
        \LWR@wrapposition; %
    }%
    (note)%
    {marginblock}%


    \setlength{\linewidth}{\LWR@templengthone}%
}


\NewDocumentEnvironment{wrapfigure}{o m o m}
{%
    \begin{LWR@setvirtualpage}*%
    \LWR@subwrapfigure{#2}{#4}%
    \renewcommand*{\@captype}{figure}%
}
{%
    \endLWR@BlockClassWP%
    \end{LWR@setvirtualpage}%
}


\NewDocumentEnvironment{wraptable}{o m o m}
{%
    \begin{LWR@setvirtualpage}*%
    \LWR@subwrapfigure{#2}{#4}%
    \renewcommand*{\@captype}{table}%
}
{%
    \endLWR@BlockClassWP%
    \end{LWR@setvirtualpage}%
}


\NewDocumentEnvironment{wrapfloat}{m o m o m}
{%
    \begin{LWR@setvirtualpage}*%
    \LWR@subwrapfigure{#3}{#5}%
    \renewcommand*{\@captype}{#1}%
}
{%
    \endLWR@BlockClassWP%
    \end{LWR@setvirtualpage}%
}

\newlength{\wrapoverhang}
```

File 563 **lwarp-wrapfig2.sty**

§ 672 Package wrapfig2

(*Emulates or patches code by* Donald Arseneau, Claudio Beccari.)

wrapfig2 (*Pkg*) wrapfig2 is emulated via a modified version of the wrapfig emulation.

for HTML output:

```
\@ifpackageloaded{color}{}{%
```

```
    \@ifpackageloaded{xcolor}{}{\LWR@origRequirePackage{xcolor}}%
}

\RequirePackage{float}

\IfPackageLoadedWithOptionsTF{wrapfig2}{WFold}
{}% v4.0
{% v5+
  \floatstyle{plain}
  \ifcsname chapter\endcsname
    \newfloat{text}{tbp}{lotx}[chapter]
  \else
    \newfloat{text}{tbp}{lotx}
  \fi
  \floatname{text}{Text}
%   \let\WF@text@caption\float@caption
}


\LWR@ProvidesPackageDrop{wrapfig2}[2022-02-16]

\LWR@origRequirePackage{lwarp-wrapfig}


\RenewDocumentEnvironment{wrapfigure}{o m o G{0pt} s}% original
  {\wrapfloat{figure}[#1]{#2}[#3]{#4}}%
  {\endwrapfloat}

\RenewDocumentEnvironment{wraptable}{o m o G{0pt} s}% original
  {\wrapfloat{table}[#1]{#2}[#3]{#4}}%
  {\endwrapfloat}

\RenewDocumentEnvironment{wrapfloat}{m o m o G{0pt}}% lwarp
{%
    \begin{LWR@setvirtualpage}*%
    \LWR@subwrapfigure{#3}{#5}%
    \renewcommand*{\@captype}{#1}%
}
{%
    \endLWR@BlockClassWP%
    \end{LWR@setvirtualpage}%
}


\IfPackageLoadedWithOptionsTF{wrapfig2}{WFold}
{% v4.0:
    \NewDocumentEnvironment{wraptext}%
        {O{l}    D||{0.5\columnwidth} D<>{0}   D(){figure}}%
    {%
        \wrapfloat{#4}[]{#1}[]{#2}%
        \tcolorbox%
    }
    {%
        \endtcolorbox%
        \endwrapfloat%
        \ignorespaces%
    }
}{}

\IfPackageLoadedWithOptionsTF{wrapfig2}{WFfive}
{% v5
```

```
    \definecolor{WFbackground}{rgb}{0.95,0.95,0.95}
    \definecolor{WFframe}{rgb}{0.1,0.1,0.1}
    \colorlet{WFtext}{black}
    \def\SetWFbgd#1{\colorlet{WFbackground}{#1}}
    \def\SetWFfrm#1{\colorlet{WFframe}{#1}}
    \def\SetWFtxt#1{\colorlet{WFtext}{#1}}
    \def\WFsplitdimens#1,#2!{\fboxrule=#1\relax\fboxsep=#2\relax}

    \NewDocumentEnvironment{wraptext}{O{0} m O{0pt} G{0.5\columnwidth}}
    {%
        \wrapfloat{text}[]{#2}[]{#4}%
    }
    {%
        \endwrapfloat%
        \ignorespaces%
    }

    \NewDocumentCommand\includeframedtext{O{\insertwidth} m O{1pt,1ex} o}%
    {%
        \WFsplitdimens #3!
        \convertcolorspec{named}{WFtext}{HTML}\LWR@tempcolor%
        \LWR@HTML@fcolorboxBlock%
            [named]{WFframe}[named]{WFbackground}{#2}%
            (%
                color:\ \LWR@origpound\LWR@tempcolor ; %
                border-radius:\ 1ex%
            )%
    }
}{% v6+
    \RequirePackage{xkeyval}

    \definecolor{WFbackground}{rgb}{0.95,0.95,0.95}
    \definecolor{WFframe}{rgb}{0.1,0.1,0.1}
    \colorlet{WFtext}{black}
    \def\SetWFbgd#1{\colorlet{WFbackground}{#1}}
    \def\SetWFfrm#1{\colorlet{WFframe}{#1}}
    \def\SetWFtxt#1{\colorlet{WFtext}{#1}}
    \def\WFsplitdimens#1,#2!{\fboxrule=#1\relax\fboxsep=#2\relax}

    \newlength{\LWR@wrapfigtwo@radius}
    \setlength{\LWR@wrapfigtwo@radius}{1ex}

    \DeclareOptionX<wraptext>{scalefactor}[0.8]{%
%       \def\WFscalefactor{#1}%
    }
    \DeclareOptionX<wraptext>{fboxrule}[1pt]{\fboxrule=#1}
    \DeclareOptionX<wraptext>{fboxsep}[1ex]{\fboxsep=#1}
    \DeclareOptionX<wraptext>{framecolor}[WFframe]{\SetWFfrm{#1}}
    \DeclareOptionX<wraptext>{backgroundcolor}[WFbackground]{\SetWFbgd{#1}}
    \DeclareOptionX<wraptext>{textcolor}[WFtext]{\SetWFtxt{#1}}
    \DeclareOptionX<wraptext>{fontstyle}[\normalfont]{#1}
    \DeclareOptionX<wraptext>{radius}[\fboxsep]{%
        \setlength{\LWR@wrapfigtwo@radius}{#1}%
    }
    \DeclareOptionX<wraptext>{insertionwidth}[0.5\columnwidth]{%
%       \insertwidth=#1%
    }

    \DeclareOptionX*{\PackageWarning{wrapfig2}{`\CurrentOption' ignored}}

```

```
    \ExecuteOptionsX<wraptext>{scalefactor, fboxrule, fboxsep, framecolor,
    backgroundcolor, textcolor, fontstyle, radius, insertionwidth}

    \ProcessOptionsX*

    \NewDocumentEnvironment{wraptext}{O{0} m O{0pt} G{0.5\columnwidth}}
    {%
        \wrapfloat{text}[]{#2}[]{#4}%
    }
    {%
        \endwrapfloat%
        \ignorespaces%
    }

    \NewDocumentCommand\includeframedtext{O{\insertwidth} m O{} o}
    {%
      \ExecuteOptionsX<wraptext>{#3}%    executes possible key=value options
        \convertcolorspec{named}{WFtext}{HTML}\LWR@tempcolor%
        \LWR@HTML@fcolorboxBlock%
            [named]{WFframe}[named]{WFbackground}%
            {\LWR@textcurrentfont{#2}}%
            (%
                color:\ \LWR@origpound\LWR@tempcolor ; %
                border-radius:\ \LWR@printlength{\LWR@wrapfigtwo@radius}%
            )%
    }
}
```

File 564 **lwarp-xbmks.sty**

§ 673 Package xbmks

xbmks (*Pkg*) xbmks is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{xbmks}[2018/07/04]

\newcommand{\xbmksetup}[1]{}
\NewDocumentCommand{\pdfbookmarkx}{o m o m}{}
\NewDocumentCommand{\currentpdfbookmarkx}{m o m}{}
\NewDocumentCommand{\subpdfbookmarkx}{m o m}{}
\NewDocumentCommand{\belowpdfbookmarkx}{m o m}{}
```

File 565 **lwarp-xcolor.sty**

§ 674 Package xcolor

(*Emulates or patches code by* Dr. Uwe Kern.)

xcolor (*Pkg*) xcolor is supported by lwarp.

§ 674.1 Limitations

\colorboxBlock and \fcolorboxBlock

\colorboxBlock and \fcolorboxBlock are provided for increased HTML compatibility, and they are identical to \colorbox and \fcolorbox in print mode. In HTML mode they place their contents into a instead of a . These s are set to display: inline-block so adjacent \colorboxBlocks appear side-by-side in HTML, although text is placed before or after each.

Print-mode definitions for \colorboxBlock and \fcolorboxBlock are created by lwarp's core if xcolor is loaded.

background: none

\fcolorbox and \fcolorboxBlock allow a background color of none, in which case only the frame is drawn, which can be useful for HTML.

color support

Color definitions, models, and mixing are fully supported without any changes required.

colored tables

\rowcolors is supported, except that the optional argument is ignored so far.

colored text and boxes

\textcolor, \colorbox, and \fcolorbox are supported.

\color and \pagecolor

\color and \pagecolor are ignored. Use CSS or \textcolor where possible.

§ 674.2 xcolor definitions: location and timing

The lwarp core and its lwarp-xcolor package are tightly integrated to allow comparable results for print, HTML, and print inside an HTML lateximage. This requires a number of definitions and redefintions depending on whether each of xcolor and lateximage is being used, and whether print or HTML is being generated. Some of these actions are one-time when xcolor is loaded, and others are temporary as lateximage is used.

When xcolor is loaded in print mode: No special actions are taken at the time that xcolor is loaded in print mode, but see \AtBeginDocument below.

When lwarp-xcolor is loaded in HTML mode: xcolor's original definitions are saved for later restoration. \LWR@restoreorigformatting is appended to restore these definitions for use inside a lateximage. New HTML-mode definitions are created for \textcolor, \pagecolor, \nopagecolor, \colorbox, \colorboxBlock, \fcolorbox, \fcolorboxBlock, and fcolorminipage.

\AtBeginDocument in print or HTML mode: See Section 89. If xcolor has been loaded, the print-mode \fcolorbox is modified to accept a background color of none, and additional definitions are created for lwarp's new macros print-mode macros \colorboxBlock, \fcolorboxBlock, and fcolorminipage. The HTML versions of these macros will already have been created by lwarp-xcolor if it has been loaded.

For use inside an HTML lateximage, \LWR@restoreorigformatting is appended to temporarily set these functions to their print-mode versions.

In a lateximage in HTML mode: \LWR@restoreorigformatting temporarily restores the print-mode definitions of xcolor's functions. See \LWR@restoreorigformatting on page 532.

\color:

Print: Used as-is.

HTML: Ignored by *pdftotext*, and will not appear.

HTML lateximage: Colors will appear in a `lateximage`.

`\textcolor`:

Print: Used as-is.

HTML: Redefined by lwarp-xcolor, page 1223.

HTML lateximage: Remembers and reuses the print version.

`\pagecolor`:

Print: Used as-is.

HTML: Ignored.

HTML lateximage: Colors will be picked up in a `lateximage`.

`\nopagecolor`:

Print: Used as-is.

HTML: Ignored.

HTML lateximage: Colors will be picked up in a `lateximage`.

`\colorbox`:

Print: Used as-is.

HTML: Redefined by lwarp-xcolor, page 1223.

HTML lateximage: Remembers and reuses the print version.

`\colorboxBlock`:

Print: Becomes `\colorbox`.

HTML: Newly defined by lwarp-xcolor to use a `<div>`, page 1224.

HTML lateximage: Remembers and reuses the print version `\colorbox`.

`\fcolorbox`:

Print: Modified to allow a background of `none`.
`\LWR@print@fcolorbox` at section 89

HTML: Redefined by lwarp-xcolor, page 1224.

HTML lateximage: Remembers and reuses the print version.

`\fcolorboxBlock`:

Print: Becomes `\fcolorbox`. Section 89

HTML: Newly defined by lwarp-xcolor to use a `<div>`, page 1225.

HTML lateximage: Remembers and reuses the print version `\fcolorbox`.

`fcolorminipage`:

Print: Newly defined in the lwarp core.
`LWR@print@fcolorminipage` at section 89

HTML: Newly defined by lwarp-xcolor, page 1225.

HTML lateximage: Uses the print version.

`\boxframe`:

Print: Used as-is.

HTML: Redefined by lwarp-xcolor, page 1226.

HTML lateximage: Remembers and reuses the print version.

§ 674.3 Package loading

for HTML output:

```
\LWR@ProvidesPackagePass{xcolor}[2016/05/11]
```

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

```
\let\color@endgroup\endgroup
```

§ 674.4 Remembering and restoring original definitions

Remember the following print-mode actions to be restored when inside a lateximage environment:

```
\LetLtxMacro\LWR@print@pagecolor\pagecolor
\LetLtxMacro\LWR@print@nopagecolor\nopagecolor
```

\LWR@restoreorigformatting Inside a lateximage the following gets restored to their print-mode actions:

```
\appto\LWR@restoreorigformatting{%
    \LetLtxMacro\pagecolor\LWR@print@pagecolor%
    \LetLtxMacro\nopagecolor\LWR@print@nopagecolor%
}
```

§ 674.5 \normalcolor

\normalcolor

```
\DeclareRobustCommand{\LWR@HTML@normalcolor}{\color{black}}%

\LWR@formatted{normalcolor}
```

§ 674.6 HTML color style

\LWR@findcurrenttextcolor Sets \LWR@tempcolor to the current color.

```
\renewcommand*{\LWR@findcurrenttextcolor}{%
    \LWR@traceinfo{LWR@findcurrenttextcolor}%
    \protect\colorlet{LWR@current@color}{.}%
    \LWR@traceinfo{LWR@findcurrenttextcolor B}%
   \protect\convertcolorspec{named}{LWR@current@color}{HTML}\LWR@tempcolor\relax%
    \LWR@traceinfo{LWR@findcurrenttextcolor: done}%
}
```

\LWR@currenttextcolorstyle Prints a color style for the current color.

```
\newcommand*{\LWR@currenttextcolorstyle}{%
    \LWR@findcurrenttextcolor%
    \ifdefstring{\LWR@tempcolor}{000000}%
    {}%
    {color: \LWR@origpound\LWR@tempcolor ; }%
}
```

\LWR@textcurrentcolor {⟨*text*⟩} Like \textcolor but uses the current \color instead.

```
\DeclareDocumentCommand{\LWR@textcurrentcolor}{m}{%
    \begingroup%
    \LWR@hook@processingtags%
    \LWR@findcurrenttextcolor%
```

```
    \InlineClass[color:\LWR@origpound\LWR@tempcolor]{textcolor}{%
        \renewcommand*{\LWR@currenttextcolor}{\LWR@origpound\LWR@tempcolor}%
        #1%
    }%
    \endgroup%
}
```

\LWR@colorstyle {⟨*1: model*⟩} {⟨*2: color*⟩}

For a color style, prints the color converted to HTML colors.

```
\NewDocumentCommand{\LWR@colorstyle}{m m}{%
    \begingroup%
    \LWR@hook@processingtags%
```

Use the xcolor package to convert to an HTML color space:

```
    \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
```

Print the converted color:

```
    \LWR@origpound\LWR@tempcolor%
    \endgroup%
}
```

\LWR@backgroundcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Similar to \textcolor, but prints black text against a color background.

Converted into an HTML hex color span.

```
\NewDocumentCommand{\LWR@backgroundcolor}{O{named} m m}{%
    \begingroup%
    \LWR@hook@processingtags%
    \InlineClass[background:\LWR@colorstyle{#1}{#2}]{backgroundcolor}{%
        #3%
    }%
    \endgroup%
}
```

§ 674.7 HTML border

\LWR@borderpadding {⟨*colorstyle*⟩} {⟨*color*⟩} Prints the HTML attributes for a color border and padding. \LWR@forceminwidth must be used first in order to set the border width.

```
\newcommand*{\LWR@borderpadding}[2]{%
   border:\LWR@printlength{\LWR@atleastonept} solid \LWR@colorstyle{#1}{#2} ; %
    padding:\LWR@printlength{\fboxsep}%
}
```

§ 674.8 High-level macros

`\color` [⟨*model*⟩] {⟨*color*⟩}

⚠ The current `\color` is used by HTML rules and frames, but does not affect the current HTML text output, due to the lack of HTML states and scoping limitations. Use `\textcolor` if possible.

```
\NewDocumentCommand{\LWR@HTML@color}{o m}{%
    \IfValueTF{#1}{%
        \LWR@print@color[#1]{#2}%
        \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
    }{%
        \LWR@print@color{#2}%
        \convertcolorspec{named}{#2}{HTML}\LWR@tempcolor%
    }%
    \edef\LWR@currenttextcolor{\LWR@origpound\LWR@tempcolor}%
}

\LWR@formatted{color}
```

`\textcolor` [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex color span.

```
\NewDocumentCommand{\LWR@HTML@textcolor}{o m m}{%
    \begingroup%
    \LWR@hook@processingtags%
    \IfValueTF{#1}{%
        \color[#1]{#2}%
    }{%
        \color{#2}%
    }%
    \InlineClass[color:\LWR@currenttextcolor]{textcolor}{#3}%
    \endgroup%
}%

\LWR@formatted{textcolor}
```

`\pagecolor` [⟨*model*⟩] {⟨*color*⟩}

Ignored. Use css instead.

```
\renewcommand*{\pagecolor}[2][named]{}
```

`\nopagecolor` Ignored.

```
\renewcommand*{\nopagecolor}{}
```

`\colorbox` [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color .

```
\NewDocumentCommand{\LWR@HTML@colorbox}{O{named} m +m}{%
    \begingroup%
```

```
    \LWR@hook@processingtags%
    \InlineClass[%
    background:\LWR@colorstyle{#1}{#2} ; %
    padding:\LWR@printlength{\fboxsep}%
    ]{colorbox}{#3}%
    \endgroup%
}
```

\colorboxBlock [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color .

```
\NewDocumentCommand{\LWR@HTML@colorboxBlock}{O{named} m +m}{%
    \begingroup%
    \LWR@hook@processingtags%

    \LWR@stoppars%

    \begin{BlockClass}[%
        background:\LWR@colorstyle{#1}{#2} ; %
        padding:\LWR@printlength{\fboxsep}%
    ]{colorboxBlock}
    #3
    \end{BlockClass}%
    \endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
    \global\booltrue{LWR@minipagethispar}%
}
```

\fcolorbox [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩}

Converted into a framed HTML hex background color span.

A background color of none creates a colored frame without a background color.

```
\NewDocumentCommand{\LWR@HTML@fcolorbox}{O{named} m O{#1} m +m}{%
    \LWR@traceinfo{HTML fcolorbox #2 #4}%
    \begingroup%
    \LWR@hook@processingtags%
    \LWR@forceminwidth{\fboxrule}%
    \ifthenelse{\equal{#4}{none}}%
        {% no background color
            \InlineClass[%
            \LWR@borderpadding{#1}{#2}%
            ]{fcolorbox}{#5}%
        }%
        {% yes background color
            \InlineClass[%
            \LWR@borderpadding{#1}{#2} ; %
            background:\LWR@colorstyle{#3}{#4}%
            ]{fcolorbox}{#5}%
        }%
    \endgroup%
}
```

\fcolorboxBlock [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩} (⟨*add'l html style*⟩)

Converted into a framed HTML hex background color span.

A background color of none creates a colored frame without a background color.

```
\NewDocumentCommand{\LWR@HTML@fcolorboxBlock}{O{named} m O{#1} m +m d()}{%
    \LWR@traceinfo{HTML fcolorboxBlock #2 #4}%
    \begingroup%
    \LWR@hook@processingtags%
    \LWR@forceminwidth{\fboxrule}%
    \LWR@stoppars%
    \ifthenelse{\equal{#4}{none}}%
        {% no background color
            \begin{BlockClass}[%
                \LWR@borderpadding{#1}{#2}%
                \IfValueT{#6}{ ; #6}%
            ]{fcolorboxBlock}
            #5
            \end{BlockClass}%
        }%
        {% yes background color
            \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
            \begin{BlockClass}[%
                background:\LWR@origpound\LWR@tempcolortwo\ ; %
                \LWR@borderpadding{#1}{#2}%
                \IfValueT{#6}{ ; #6}%
            ]{fcolorboxBlock}
            #5
            \end{BlockClass}%
        }%
    \endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
    \global\booltrue{LWR@minipagethispar}%
    \LWR@traceinfo{HTML fcolorboxBlock done}%
}
```

Creates a framed HTML around its contents.

A print-output version is defined in the lwarp core: section 89

\LWR@subfcolorminipage {⟨*framemodel*⟩} {⟨*framecolor*⟩} {⟨*background tag*⟩} {⟨*height*⟩}

```
\NewDocumentCommand{\LWR@subfcolorminipage}{m m m m}{%
    \LWR@stoppars%
    \begin{BlockClass}[%
        #3%
        \LWR@borderpadding{#1}{#2} ; %
        \IfValueT{#4}{height:\LWR@printlength{\LWR@tempheight} ; }%
        width:\LWR@printlength{\LWR@tempwidth}%
```

```
    ]{fcolorminipage}%
}
```

`fcolorminipage` (*env.*) `[`⟨*1:framemodel*⟩`]` `{`⟨*2:framecolor*⟩`}` `[`⟨*3:boxmodel*⟩`]` `{`⟨*4:boxcolor*⟩`}` `[`⟨*5:align*⟩`]` `[`⟨*6:height*⟩`]` `[`⟨*7:inner-align*⟩`]` `{`⟨*8:width*⟩`}`

```
\NewDocumentEnvironment{LWR@HTML@fcolorminipage}{O{named} m O{#1} m O{c} o o m}
{%
    \LWR@hook@processingtags%
    \setlength{\LWR@tempwidth}{#8}%
    \IfValueT{#6}{\setlength{\LWR@tempheight}{#6}}%
    \LWR@forceminwidth{\fboxrule}%
    \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
    \ifthenelse{\equal{#4}{none}}%
        {\LWR@subfcolorminipage{#1}{#2}{}{#6}}%
        {%
            \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
            \LWR@subfcolorminipage{#1}{#2}%
                {background:\LWR@origpound\LWR@tempcolortwo\ ; }%
                {#6}%
        }%
}%
{%
    \end{BlockClass}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
    \global\booltrue{LWR@minipagethispar}%
}
```

`\boxframe` `{`⟨*width*⟩`}` `{`⟨*height*⟩`}` `{`⟨*depth*⟩`}`

The depth is added to the height, but the box is not decended below by the depth. `\textcolor` is honored.

```
\newcommand*{\LWR@HTML@boxframe}[3]{%
    {%
        \setlength{\LWR@tempwidth}{#1}%
        \setlength{\LWR@tempheight}{#2}%
        \addtolength{\LWR@tempheight}{#3}%
        \LWR@forceminwidth{\fboxrule}%
        \LWR@findcurrenttextcolor%
        \InlineClass[%
            display:inline-block ; %
            border:%
                \LWR@printlength{\LWR@atleastonept} % space
                solid % space
                \LWR@currenttextcolor{} ; % space
            width:\LWR@printlength{\LWR@tempwidth} ; %
            height:\LWR@printlength{\LWR@tempheight}%
        ]{boxframe}{}%
    }%
}

\LWR@formatted{boxframe}
```

§ 674.9 Row colors

\rowc@l@rs [⟨*cmds*⟩] {⟨*startrow*⟩} {⟨*odd color*⟩} {⟨*even color*⟩}

```
\newcommand*{\LWR@xcolortempcolor}{}

\def\rowc@l@rs[#1]#2#3#4%
{%
   \rownum=1%
   \@rowcolorstrue%
   \@ifxempty{#3}%
     {\def\@oddrowcolor{\@norowcolor}}%
     {%
        \convertcolorspec{named}{#3}{HTML}\LWR@xcolortempcolor%
        \edef\@oddrowcolor{%
            \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%
        }%
     }%
   \@ifxempty{#4}%
     {\def\@evenrowcolor{\@norowcolor}}%
     {%
        \convertcolorspec{named}{#4}{HTML}\LWR@xcolortempcolor%
        \edef\@evenrowcolor{%
            \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%
        }%
     }%
   \if@rowcmd
     \def\@rowcolors
     {%
%          #1%
        \if@rowcolors
%         \noalign{%
            \relax\ifnum\rownum<#2\@norowcolor\else
            \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi\fi%
%          }%
        \fi%
     }%
   \else
     \def\@rowcolors
     {%
        \if@rowcolors
            \ifnum\rownum<#2%
%            \noalign{%
               \@norowcolor
%             }
            \else
%            #1%
%            \noalign{%
               \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi%
%             }%
            \fi
        \fi%
     }%
   \fi
   \ignorespaces%
}
```

\@norowcolor Turns off color for this row.

```
\def\@norowcolor{%
    \renewcommand{\LWR@xcolorrowHTMLcolor}{}%
```

```
}
```

\@rowc@lors Executed at the end of each row.

```
\def\@rowc@lors{%
%   \noalign{%
        \advance\rownum\@ne%
%   }%
    \@rowcolors%
}
```

File 566 **lwarp-xechangebar.sty**

§ 675 Package

xechangebar

xechangebar (*Pkg*) xechangebar is ignored

for HTML output:

```
\LWR@ProvidesPackageDrop{xechangebar}[2017/08/03]
\LWR@origRequirePackage{lwarp-changebar}
```

File 567 **lwarp-xellipsis.sty**

§ 676 Package

xellipsis

(*Emulates or patches code by* Donald P. Goodman III.)

xellipsis (*Pkg*) xellipsis is patched for use by lwarp.

When non-zero, each of the spaces is converted to an HTML thin unbreakable space.

for HTML output:

```
\LWR@ProvidesPackagePass{xellipsis}[2015/11/01]

\newcommand*{\LWR@xellipsespace}[1]{%
\ifdim#1=0pt\else%
    \ifdim#1<\fontdimen2\font%
        \,%
    \else%
        ~%
    \fi%
\fi%
}

\def\xelip{%
\mbox{%
    \LWR@xellipsespace{\xelipprebef}%
    \xelipprechar%
    \LWR@xellipsespace{\xelippreaft}%
    \LWR@xellipsespace{\xelipbef}%
    \xelipchar%
    \xel@loopi = 1%
    \loop\ifnum\xelipnum>\xel@loopi%
        \advance\xel@loopi by1%
        \LWR@xellipsespace{\xelipgap}%
```

```
        \xelipchar%
    \repeat%
    \LWR@xellipsespace{\xelipaft}%
    \LWR@xellipsespace{\xelippostbef}%
    \xelippostchar%
    \LWR@xellipsespace{\xelippostaft}%
}%
}%
```

File 568 **lwarp-xetexko.sty**

§ 677 Package xetexko

(*Emulates or patches code by* Dohyun Kim.)

xetexko (*Pkg*) xetexko is patched for use by lwarp.

for HTML output:

```
\LWR@loadbefore{xetexko}

\LWR@ProvidesPackagePass{xetexko}[2021/09/06]

\protected\def\typesetvertical{}
\protected\def\typesethorizontal{}

\def\verticaltypesetting{\BlockClass{verticalrl}}
\def\beginverticaltypesetting{\BlockClass{verticalrl}}
\def\endverticaltypesetting{\endBlockClass}

\protected\def\vertical#1{\BlockClass{verticalrl}}
\protected\def\endvertical{\endBlockClass}
\protected\def\horizontal#1{\BlockClass{horizontaltb}}
\protected\def\endhorizontal{\endBlockClass}
\DeclareDocumentCommand{\vertlatin}{m}{#1}
```

File 569 **lwarp-xevlna.sty**

§ 678 Package xevlna

(*Emulates or patches code by* Zdeněk Wagner.)

xevlna (*Pkg*) xevlna is patched for use by lwarp.

Non-breakable spaces are inserted into HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{xevlna}[2016/09/05]

\def\ProcessCSpreposition{\ifx\next\xevlnaXeTeXspace\HTMLentity{nbsp}\fi}

\appto{\LWR@hook@processingtags}{\xevlnaDisable}%
```

File 570 **lwarp-xfakebold.sty**

§ 679 Package
# xfakebold

(*Emulates or patches code by* Herbert Voss.)

xfakebold (*Pkg*) xfakebold is patched for use by lwarp, and additional underlying support is found in the lwarp core.

⚠ text mode xfakebold is only used in SVG math and `latexi`mages. Text mode is not set bold, but `\setBold` in text will be applied to any following SVG math.

for HTML output:

```
\LWR@ProvidesPackagePass{xfakebold}[2020/06/24]

\newcommand*{\LWR@HTML@setBold}{\booltrue{LWR@xfakebold}}
\LWR@formatted{setBold}

\newcommand*{\LWR@HTML@unsetBold}{\boolfalse{LWR@xfakebold}}
\LWR@formatted{unsetBold}

\renewcommand*{\LWR@applyxfakebold}{%
    \ifbool{LWR@xfakebold}{\LWR@print@setBold}{\LWR@print@unsetBold}%
}
```

For MathJax, xfakebold is ignored.

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\setBold}[1][]{}}
\CustomizeMathJax{\newcommand{\unsetBold}{}}
\end{warpMathJax}
```

File 571 **lwarp-xfrac.sty**

§ 680 Package
# xfrac

(*Emulates or patches code by* The LaTeX3 Project.)

xfrac (*Pkg*) Supported by adding xfrac instances, and emulated for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{xfrac}[2018-08-23]
```

⚠ font size In the user's document preamble, lwarp should be loaded after font-related setup. During HTML conversion, this font is used by lwarp to generate its initial PDF output containing HTML tags, later to be converted by *pdftotext* to a plain text file. While the text may be in any font which *pdftotext* can read, the math is directly converted into SVG images using this same user-selected font. `xfrac` below is set for the Latin Modern (lmr) font. If another font is used, it may be desirable to redefine `\xfracHTMLfontsize` with a different em size.

`\sfrac` `[`⟨*instance*⟩`]` `{`⟨*num*⟩`}` `[`⟨*sep*⟩`]` `{`⟨*denom*⟩`}`

A text-mode instance for the default font is provided below. The numerator and denominator formats are adjusted to encase everything in HTML tags. `\scalebox`

is made null inside the numerator and denominator, since the HTML tags should not be scaled, and we do not want to introduce additional HTML tags for scaling.

In math mode, which will appear inside a `lateximage`, no adjustments are necessary.

`\xfracHTMLfontsize` User-redefinable macro which controls the font size of the fraction.

```
\newcommand*{\xfracHTMLfontsize}{.6em}
```

instances Instances of xfrac for various font choices:

Produce CSS for a small raised numerator and a small denominator.

Scaling is turned off so that *pdftotext* correctly reads the result.

```
\DeclareInstance{xfrac}{default}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}

\DeclareInstance{xfrac}{lmr}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}

\DeclareInstance{xfrac}{lmss}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
```

```
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}
\DeclareInstance{xfrac}{lmtt}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\LWRsfrac}[2][/]{{}^\LWRsfracnumerator\!#1{}_{#2}}}
\CustomizeMathJax{\newcommand{\sfrac}[2][]{\def\LWRsfracnumerator{#2}\LWRsfrac}}
\end{warpMathJax}
```

File 572 **lwarp-xltabular.sty**

§ 681 Package

xltabular

(*Emulates or patches code by* Rolf Niepraschk, Herbert Voss.)

`xltabular` (*Pkg*) xltabular is emulated by lwarp.

for HTML output: Relies on tabularx.

⚠ table numbering At present, an `xltabular` without a caption or with only a `\caption*` may be misnumbered in HTML, so it may be necessary to place at the end of the table:

```
\warpHTMLonly{\addtocounter{table}{-1}}
```

```
\RequirePackage{tabularx}
```

```
\RequirePackage{ltablex}

\LWR@ProvidesPackageDrop{xltabular}[2018/05/23]

\DeclareDocumentEnvironment{xltabular}{o m m}
{\longtable{#3}}
{\endlongtable}
```

File 573 **lwarp-xltxtra.sty**

§ 682 Package xltxtra

(*Emulates or patches code by* Will Robertson, Jonathan Kew.)

xltxtra (*Pkg*) xltxtra is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{xltxtra}[2016/01/21]

\RequirePackage{realscripts}
\RequirePackage{metalogo}
\newcommand*\TeX@logo@spacing[6]{}

\newcommand*{\vfrac}[2]{%
\textsuperscript{#1}/\textsubscript{#2}%
}

\newcommand\namedglyph[1]{%
  \@tempcnta=\XeTeXglyphindex "#1"\relax
  \ifnum\@tempcnta>0
    \XeTeXglyph\@tempcnta
  \else
    \xxt@namedglyph@fallback{#1}%
  \fi}

\newcommand\xxt@namedglyph@fallback[1]{[#1]}

\DeclareDocumentCommand{\showhyphens}{m}{}
```

File 574 **lwarp-xmpincl.sty**

§ 683 Package xmpincl

(*Emulates or patches code by* Maarten Sneep.)

xmpincl (*Pkg*) xmpincl is ignored.

for HTML output: Discard all options for lwarp-xmpincl:

```
\LWR@ProvidesPackageDrop{xmpincl}[2008/05/10]

\newcommand*{\includexmp}[1]{}
```

File 575 **lwarp-xpiano.sty**

§ 684 Package xpiano

(*Emulates or patches code by* Enrico Gregorio.)

xpiano (*Pkg*) xpiano is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{xpiano}

\ExplSyntaxOn
\NewDocumentCommand{\LWR@print@keyboard}{ O{}m }
{
\xpiano_keyboard:nn { #1 } { #2 }
}

\NewDocumentCommand{\LWR@HTML@keyboard}{ O{}m }
{
\begin{lateximage}*
    [%
        -xpiano-~\PackageDiagramAltText{}: \detokenize\expandafter{#2}%
    ]
    [\detokenize\expandafter{#1}]
\xpiano_keyboard:nn { #1 } { #2 }
\end{lateximage}
}
\ExplSyntaxOff

\LWR@formatted{keyboard}
```

File 576 **lwarp-xpinyin.sty**

§ 685 Package xpinyin

(*Emulates or patches code by* Soben Lee.)

xpinyin (*Pkg*) xpinyin is supported.

Pinyin is disabled for file names, the sidetoc, and regular footnotes, but is left enabled for minipage footnotes, as per the print mode.

for HTML output:

```
\LWR@ProvidesPackagePass{xpinyin}[2019-04-07]
```

The original's boxes are not used, instead the contents are used with <ruby>, <rt>, and <rp> tags per modern HTML. Color is detected. ratio is ignored for *pdftotext* to work correctly. Extra spaces are placed inside the tags to allow line breaks in the HTML text.

```
\ExplSyntaxOn
\cs_new_protected_nopar:Npn \LWR@HTML@__xpinyin_make_pinyin_box:nnn #1#2#3
{
    \color_group_begin: \color_ensure_current:
```

```
    \l__xpinyin_pinyin_box_hook_tl
    \renewcommand*{\l__xpinyin_ratio_tl}{1}% for pdftotext
    \__xpinyin_select_font:
    \clist_if_exist:cTF { c__xpinyin_multiple_ #1 _clist }
        { \l__xpinyin_multiple_tl \l__xpinyin_format_tl }
        { \l__xpinyin_format_tl }
    \ifdefempty{\l__xpinyin_format_tl}
        {#3}
        {\LWR@textcurrentcolor{#3}}
    \color_group_end:
}
\LWR@formatted{__xpinyin_make_pinyin_box:nnn}

\cs_new_protected_nopar:Npn \LWR@HTML@__xpinyin_CJKsymbol:nn #1#2
  {
    \__xpinyin_leavevmode:
    \LWR@htmltagc{ruby}
    \__xpinyin_save_CJKsymbol:n {#2}\null% \null removes extra space
    \LWR@htmltagc{rp}(\LWR@htmltagc{/rp\space}
    \LWR@htmltagc{rt}
   \__xpinyin_make_pinyin_box:nnn {#1} {#2} { \use:c { c__xpinyin_ #1 _tl } }
    \LWR@htmltagc{/rt\space}
    \LWR@htmltagc{rp})\LWR@htmltagc{/rp\space}
    \LWR@htmltagc{/ruby\space}\null
  }
\LWR@formatted{__xpinyin_CJKsymbol:nn}

\cs_new_protected_nopar:Npn \LWR@HTML@__xpinyin_single_CJKsymbol:nn #1#2
  {
    \__xpinyin_leavevmode:
    \LWR@htmltagc{ruby}
    \__xpinyin_save_CJKsymbol:n {#1}\null% \null removes extra space
    \LWR@htmltagc{rp}(\LWR@htmltagc{/rp\space}
    \LWR@htmltagc{rt}
    \__xpinyin_make_pinyin_box:xnn
      { \__xpinyin_to_unicode:n {#1} } {#1} { \__xpinyin_pinyin:n {#2} }
    \LWR@htmltagc{/rt\space}
    \LWR@htmltagc{rp})\LWR@htmltagc{/rp\space}
    \LWR@htmltagc{/ruby\space}\null
  }
\LWR@formatted{__xpinyin_single_CJKsymbol:nn}

\ExplSyntaxOff
```

The lwarp core uses the following to disable CJK xpinyin for filenames, sideTOC, and footnotes.

```
\renewcommand*{\LWR@disablepinyin}{\disablepinyin}

\FilenameNullify{\LWR@disablepinyin}
```

File 577 **lwarp-xr.sty**

§ 686 Package **xr**

(*Emulates or patches code by* Jean-Pierre Drucbert, David Carlisle.)

xr (*Pkg*) xr is patched for use by lwarp. The `*_html.aux` file is used. `\externaldocument` is modified to also accept the optional arguments for xr-hyper, which currently uses xr for HTML output.

See section 5.17.

for HTML output:

```
\LWR@ProvidesPackagePass{xr}[2019/07/22]%

\LetLtxMacro\LWR@orig@externaldocument\externaldocument

\RenewDocumentCommand{\externaldocument}{O{} O{} m O{}}{%
    \ifblank{#1}{%
        \LWR@orig@externaldocument{#3_html}%
    }{%
        \LWR@orig@externaldocument[#1]{#3_html}%
    }%
}
```

File 578 **lwarp-xr-hyper.sty**

§ 687 Package xr-hyper

(*Emulates or patches code by* David Carlisle.)

xr-hyper (*Pkg*) xr-hyper is replaced by xr, which is modified to accept the optional arguments for `\externaldocument`. So far, no hyperlinks are provided for citations.

See section 5.17.

for HTML output:

```
\LWR@ProvidesPackageDrop{xr-hyper}[2019/10/03]%

\LWR@origRequirePackage{lwarp-xr}
```

File 579 **lwarp-xtab.sty**

§ 688 Package xtab

(*Emulates or patches code by* Peter Wilson.)

xtab (*Pkg*) xtab is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{xtab}[2011/07/31]
```

⚠ Misplaced alignment tab character & For `\tablefirsthead`, etc., enclose them as follows:

```
\StartDefiningTabulars
\tablefirsthead
. . .
\StopDefiningTabulars
```

See section 8.10.1.

⚠ lateximage supertabular and xtab are not supported inside a `lateximage`.

```
\newcommand{\LWRXT@firsthead}{}

\newcommand{\tablefirsthead}[1]{%
    \long\gdef\LWRXT@firsthead{#1}%
}

\newcommand{\tablehead}[1]{}

\newcommand{\tablelasthead}[1]{}

\newcommand{\notablelasthead}{}

\newcommand{\tabletail}[1]{}

\newcommand{\LWRXT@lasttail}{}

\newcommand{\tablelasttail}[1]{%
    \long\gdef\LWRXT@lasttail{#1}%
}


\newcommand{\tablecaption}[2][]{%
    \long\gdef\LWRXT@caption{%
        \ifblank{#1}%
            {\caption{#2}}%
            {\caption[#1]{#2}}%
    }%
}

\let\topcaption\tablecaption
\let\bottomcaption\tablecaption


\newcommand*{\LWRXT@caption}{}

\newcommand*{\shrinkheight}[1]{}

\newcommand*{\xentrystretch}[1]{}

\NewDocumentEnvironment{xtabular}{s o m}
{%
\LWR@traceinfo{xtabular}%
\table%
\LWRXT@caption%
\begin{tabular}{#3}%
\TabularMacro\ifdefvoid{\LWRXT@firsthead}%
{\LWR@getmynexttoken}%
{\expandafter\LWR@getmynexttoken\LWRXT@firsthead}%
}%
{%
\ifdefvoid{\LWRXT@lasttail}%
{}%
{%
\TabularMacro\ResumeTabular%
\LWRXT@lasttail%
}%
\end{tabular}%
\endtable%


\gdef\LWRXT@caption{}%
```

```
\LWR@traceinfo{xtabular done}%
}

\NewDocumentEnvironment{mpxtabular}{s o m}
{\minipage{\linewidth}\xtabular{#3}}
{\endxtabular\endminipage}
```

File 580 **lwarp-xunicode.sty**

§ 689 Package

xunicode

xunicode (*Pkg*) Error if xunicode is loaded after lwarp.

Patch lwarp-xunicode, but also verify that is was loaded before lwarp:

for HTML output:

```
\LWR@loadbefore{xunicode}%

\LWR@ProvidesPackagePass{xunicode}[2011/09/09]
```

\textcircled becomes a span with a rounded border. \providecommand is used to avoid conflict with textcomp.

```
\providecommand*{\LWR@HTML@textcircled}[1]{%
    \InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
}

\LWR@formatted{textcircled}
```

Nullify xunicode macros when generating filenames:

```
\FilenameNullify{%
    \renewcommand*{\textdegree}{}%
    \renewcommand*{\textcelsius}{}%
    \renewcommand*{\textohm}{}%
    \renewcommand*{\textmu}{}%
    \renewcommand*{\textlquill}{}%
    \renewcommand*{\textrquill}{}%
    \renewcommand*{\textcircledP}{}%
    \renewcommand*{\texttwelveudash}{}%
    \renewcommand*{\textthreequartersemdash}{}%
    \renewcommand*{\textmho}{}%
    \renewcommand*{\textnaira}{}%
    \renewcommand*{\textpeso}{}%
    \renewcommand*{\textrecipe}{}%
    \renewcommand*{\textinterrobang}{}%
    \renewcommand*{\textinterrobangdown}{}%
    \renewcommand*{\textperthousand}{}%
    \renewcommand*{\textpertenthousand}{}%
    \renewcommand*{\textbaht}{}%
    \renewcommand*{\textdiscount}{}%
    \renewcommand*{\textservicemark}{}%
    \renewcommand*{\textcircled}[1]{#1}%
    \renewcommand*{\capitalcedilla}[1]{#1}%
    \renewcommand*{\capitalogonek}[1]{#1}%
    \renewcommand*{\capitalgrave}[1]{#1}%
    \renewcommand*{\capitalacute}[1]{#1}%
    \renewcommand*{\capitalcircumflex}[1]{#1}%
```

```
    \renewcommand*{\capitaltilde}[1]{#1}%
    \renewcommand*{\capitaldieresis}[1]{#1}%
    \renewcommand*{\capitalhungarumlaut}[1]{#1}%
    \renewcommand*{\capitalring}[1]{#1}%
    \renewcommand*{\capitalcaron}[1]{#1}%
    \renewcommand*{\capitalbreve}[1]{#1}%
    \renewcommand*{\capitalmacron}[1]{#1}%
    \renewcommand*{\capitaldotaccent}[1]{#1}%
}% FilenameNullify
```

File 581 **lwarp-xurl.sty**

§ 690 Package xurl

xurl (*Pkg*) xurl is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{xurl}[2020/01/14]

\def\useOriginalUrlSetting{}
```

File 582 **lwarp-xy.sty**

§ 691 Package xy

(*Emulates or patches code by* Kristoffer H. Rose, Ross Moore.)

xy (*Pkg*) xy is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{xy}[2013/10/06]
```

After xy modules have been loaded:

```
\AtBeginDocument{
```

The original definitions without a `lateximage`:

```
\LetLtxMacro\LWR@orig@xy\xy
\LetLtxMacro\LWR@orig@endxy\endxy
```

The outer-most `xy` environment is placed in a `lateximage`, but not more than one level deep, which would conflict with xy:

```
\renewcommand*{\xy}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {\addtocounter{LWR@lateximagedepth}{1}}%
        {\begin{lateximage}[-xy-~\PackageDiagramAltText]}%
    \LWR@orig@xy%
}

\renewcommand*{\endxy}{%
    \LWR@orig@endxy%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{1}%
        {\addtocounter{LWR@lateximagedepth}{-1}}%
        {\end{lateximage}}%
}
```

The \xybox must use the original definitions of \xy, \endxy:

```
\def\xybox#1{%
    \LWR@orig@xy#1\LWR@orig@endxy%
    \Edge@c={\rectangleEdge}\computeLeftUpness@%
}
```

If \xygraph is used, it is placed inside a lateximage:

```
\@ifundefined{xygraph}{}{
\LetLtxMacro\LWR@origxygraph\xygraph
\renewcommand{\xygraph}[1]{%
    \begin{lateximage}[-xy- xygraph \PackageDiagramAltText]
    \LWR@origxygraph{#1}
    \end{lateximage}
}
}% xygraph defined
}% AtBeginDocument
```

File 583 **lwarp-zhlineskip.sty**

§ 692 Package zhlineskip

zhlineskip (*Pkg*) zhlineskip is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{zhlineskip}[2019/05/15]
\newcommand*\SetTextEnvironmentSinglespace[1]{}
\newcommand*\RestoreTextEnvironmentLeading[1]{}
\newcommand*\SetMathEnvironmentSinglespace[1]{}
\newcommand*\RestoreMathEnvironmentLeading[1]{}
```

File 584 **lwarp-zwpagelayout.sty**

§ 693 Package zwpagelayout

(*Emulates or patches code by* Zdeněk Wagner.)

zwpagelayout (*Pkg*) zwpagelayout is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{zwpagelayout}[2013/01/13]
\def\noBboxes{}
\@onlypreamble\noBboxes

\expandafter\ifx\csname definecolor\endcsname\relax \else
  \definecolor{cmykblack}{cmyk}{0,0,0,1}
  \definecolor{grblack}{gray}{0}
%   \ifzwpl@redefineblack
%     \definecolor{black}{cmyk}{0,0,0,1}\color{black}
```

```
%   \fi
  \definecolor{cmykred}{cmyk}{0,1,1,0}
  \definecolor{cmykgreen}{cmyk}{1,0,1,0}
  \definecolor{cmykblue}{cmyk}{1,1,0,0}
  \definecolor{rgbred}{rgb}{1,0,0}
  \definecolor{rgbgreen}{rgb}{0,1,0}
  \definecolor{rgbblue}{rgb}{0,0,1}
%   \ifzwpl@redefinetocmyk
%     \definecolor{red}{cmyk}{0,1,1,0}
%     \definecolor{green}{cmyk}{1,0,1,0}
%     \definecolor{blue}{cmyk}{1,1,0,0}
%   \fi
\fi

\let\OverprintXeTeXExtGState\relax

\DeclareRobustCommand\SetOverprint{\ignorespaces}
\DeclareRobustCommand\SetKnockout{\ignorespaces}
\DeclareRobustCommand\textoverprint[1]{{\SetOverprint#1}}
\DeclareRobustCommand\textknockout[1]{{\SetKnockout#1}}

\def\SetPDFminorversion#1{}
\@onlypreamble\SetPDFminorversion

\newcommand*\Vcorr{}

\DeclareRobustCommand\vb[1][]{}
\NewDocumentCommand{\NewOddPage}{* o}{}
\NewDocumentCommand{\NewEvenPage}{* o}{}
\def\SetOddPageMessage#{\gdef\ZW@oddwarning}
\def\SetEvenPageMessage#{\gdef\Z@@evenwarning}
\def\ZW@oddwarning{Empty page inserted}\let\ZW@evenwarning\ZW@oddwarning

\def\clap#1{#1}

\def\CropFlap{2in}
\def\CropSpine{1in}
\def\CropXSpine{1in}
\def\CropXtrim{.25in}
\def\CropYtrim{.25in}
\def\UserWidth{5in}
\def\UserLeftMargin{1in}
\def\UserRightMargin{1in}
\def\UserTopMargin{1in}
\def\UserBotMargin{1in}
\def\thePageNumber{\LWR@origpound\,\arabic{page}}
\ifXeTeX
\def\ifcaseZWdriver{\ifcase2}
\else
\def\ifcaseZWdriver{\ifcase1}
\fi
\DeclareRobustCommand\ZWifdriver[2]{}
```

File 585 **lwarp-patch-komascript.sty**

§ 694 Package

patch-komascript

`lwarp-patch-komascript` (*Pkg*) Patches for komascript classes.

lwarp loads this package when scrbook, scrartcl, or scrreprt classes are detected.

Many features are ignored during the HTML conversion. The goal is source-level compatibility.

`\captionformat`, `\figureformat`, and `\tableformat` are not yet emulated.

⚠ Not fully tested! Please send bug reports!

Some features have not yet been tested. Please contact the author with any bug reports.

for HTML output:

```
\ProvidesPackage{lwarp-patch-komascript}
```

typearea is emulated.

```
\RequirePackage{lwarp-typearea}
```

tocbasic is emulated.

```
\RequirePackage{lwarp-tocbasic}
```

scrextend patches most of the new macros.

```
\RequirePackage{lwarp-scrextend}
```

Indexing macros, simplified for lwarp:

```
\AtBeginDocument{

\renewcommand*{\idx@heading}{%
  \idx@@heading{\indexname}%
}

\renewenvironment{theindex}{%
  \idx@heading%
  \index@preamble\par\nobreak
    \LetLtxMacro\item\LWR@indexitem%
    \LetLtxMacro\subitem\LWR@indexsubitem%
    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
}
{}

\renewcommand*\indexspace{}

}% AtBeginDocument
```

The `\minisec` is placed inside a `<div>` of class `minisec`.

```
\renewcommand*{\minisec}[1]{
    \begin{BlockClass}{minisec}
    #1
    \end{BlockClass}
}
```

The part and chapter preambles are placed as plain text just after each heading.

```
\@ifundefined{setpartpreamble}{}{
```

```
\RenewDocumentCommand{\setpartpreamble}{o o +m}{%
    \renewcommand{\part@preamble}{#3}%
}
}

\@ifundefined{setchapterpreamble}{}{
\RenewDocumentCommand{\setchapterpreamble}{o o +m}{%
    \renewcommand{\chapter@preamble}{#3}%
}
}
```

Do not use \chaptername:

```
\renewcommand*{\LWR@printchaptername}{}
```

Simple captions are used in all cases.

```
\AtBeginDocument{
\AtBeginDocument{
    \LetLtxMacro\captionbelow\caption
    \LetLtxMacro\captionabove\caption

    \LetLtxMacro\captionofbelow\captionof
    \LetLtxMacro\captionofabove\captionof
}
}

\RenewDocumentEnvironment{captionbeside}{o m o o o s}
{}
{%
    \IfValueTF{#1}%
        {\caption[#1]{#2}}%
        {\caption{#2}}%
}

\RenewDocumentEnvironment{captionofbeside}{m o m o o o s}
{}
{%
    \IfValueTF{#2}%
        {\captionof{#1}[#2]{#3}}%
        {\captionof{#1}{#3}}%
}

\RenewDocumentCommand{\setcapindent}{s m}{}
\renewcommand*{\setcaphanging}{}
\renewcommand*{\setcapwidth}[2][]{}
\renewcommand*{\setcapdynwidth}[2][]{}
\RenewDocumentCommand{\setcapmargin}{s o m}{}
```

File 586 **lwarp-patch-memoir.sty**

§ 695 Package patch-memoir

(*Emulates or patches code by* Peter Wilson.)

lwarp-patch-memoir (*Pkg*) Patches for memoir class.

⚠ Not fully tested! Please send bug reports!

lwarp loads this package when the memoir class is detected.

⚠ captions lwarp uses caption, which causes a warning from memoir. This is normal. Adjust captions via caption, instead of memoir.

⚠ options clash While emulating memoir, lwarp pre-loads a number of packages (section 695.1). This can cause an options clash when the user's document later loads the same packages with options. To fix this problem, specify the options before loading lwarp:

```
\documentclass{memoir}
. . .
\PassOptionsToPackage{options_list}{package_name}
. . .
\usepackage{lwarp}
. . .
\usepackage{package_name}
```

⚠ version numbers memoir emulates a number of packages, and declares a version date for each which often does not match the date of the corresponding freestanding package. This can cause warnings about incorrect version numbers. Since lwarp is intended to support the freestanding packages, which are often newer than the date declared by memoir, it is hoped that memoir will update and change its emulated version numbers to match.

`\label(bookmark){tag}` `\label` accepts an optional `(bookmark)` argument, but this is ignored in HTML.

⚠ `comment` The `comment` environment is from the comment package, and thus requires that the `\begin` and `\end` each be on its own line:

```
\begin{comment}
This is a comment.
\end{comment}
```

`\newcomment` Comments defined with `\newcomment` use memoir's defintions, and behave as expected, where the `\begin` and `\end` do have to each be on its own line.

⚠ verbatim footnotes `\verbfootnote` is not supported.

⚠ `\newfootnoteseries` `\newfootnoteseries`, etc. are not supported.

⚠ page notes lwarp loads pagenote to perform memoir's pagenote functions, but there are minor differences in `\pagenotesubhead` and related macros.

page notes with cleveref To add support for pagenotes with cleveref, add:

```
\crefname{pagenote}{page note}{page notes}
\Crefname{pagenote}{Page note}{Page notes}
```

page note `\nameref` Note that for print mode, `\nameref` print the section name where the page notes are declared in the text, but for HTML it prints the name where the page notes are printed.

⚠ poems Poem numbering is not supported.

⚠ `verbatim` The `verbatim` environment does not yet support the memoir enhancements. It is currently recommended to load and use fancyvrb instead.

⚠ glossaries The memoir glossary system is not yet supported by *lwarpmk*. The glossaries package may be used instead, but does require the glossary entries be changed from the memoir syntax to the glossaries syntax.

for HTML output:
```
\ProvidesPackage{lwarp-patch-memoir}
```

§ 695.1 Packages

These are pre-loaded to provide emulation for many of memoir's functions. memoir pretends that abstract, etc. are already loaded, via its "emulated" package mechanism, but lwarp is directly loading the "lwarp-" version of each, which happens to avoid memoir's emulation system.

```
\RequirePackage{lwarp-abstract}% req'd
% \RequirePackage{lwarp-array}% no longer req'd
\RequirePackage{lwarp-booktabs}% req'd
% \RequirePackage{lwarp-ccaption}% emualated below
\RequirePackage{lwarp-changepage}% req'd
\RequirePackage{lwarp-crop}
% \RequirePackage{lwarp-dcolumn}% no longer req'd
\RequirePackage{lwarp-enumerate}% req'd
\RequirePackage{lwarp-epigraph}% req'd
\RequirePackage{lwarp-fancyvrb}% req'd
\RequirePackage{lwarp-footmisc}% req'd


\let\framed\relax \let\endframed\relax
\let\shaded\relax \let\endshaded\relax
\let\leftbar\relax \let\endleftbar\relax
\let\snugshade\relax \let\endsnugshade\relax
\RequirePackage{lwarp-framed}% req'd

\RequirePackage{lwarp-hanging}% req'd
\RequirePackage{lwarp-makeidx}% req'd
\DisemulatePackage{moreverb}
\RequirePackage{lwarp-moreverb}
\RequirePackage{lwarp-mparhack}
\RequirePackage{lwarp-needspace}% req'd
\RequirePackage{lwarp-nextpage}% req'd
\RequirePackage{lwarp-pagenote}% req'd
\RequirePackage{lwarp-parskip}
\RequirePackage{lwarp-setspace}% req'd
\RequirePackage{lwarp-showidx}


\makeindex

% \RequirePackage{lwarp-tabularx}% no longer req'd
\RequirePackage{lwarp-titling}% req'd
% \RequirePackage{lwarp-tocbibind}% not emulated by memoir
\RequirePackage{lwarp-tocloft}% req'd
\RequirePackage{lwarp-verse}% req'd
```

§ 695.2 Label handling

Insert the lwarp label mechanism into the memoir package mechanism:

- `\@mem@old@label` is the LaTeX definition of `\label`.
- `\LWR@orig@label` becomes the memoir definition.
- lwarp's `\LWR@new@label` uses `\LWR@orig@label`.
- Want memoir's `\label` to use lwarp's `\label`, which then would use LaTeX's `\label`.
- So:
  - `\@mem@old@label` is set to `\LWR@new@label`.
  - `\LWR@orig@label` is set to `\@mem@old@label`.
- cleveref then encapsulates all the above with `\cref@old@label`.
- For a subcaption, cleveref modifies memoir's `\sf@memsub@label`, but that change is undone by lwarp.

```
\LetLtxMacro\LWR@orig@label\@mem@old@label
\LetLtxMacro\@mem@old@label\LWR@new@label
```

Patches for subfloats to support additional lwarp labels. This is the non-hyperref version from memoir.

```
\AtBeginDocument{
    \renewcommand*{\sf@@memsub@label}[1]{%
        \@bsphack
        \sf@@memsub@label@hook{#1}%
%         \@memoldlabel{#1}%
        \cref@label{#1}%                    lwarp
        \LWR@label@createtag{sub@#1}%           lwarp
        \protected@write\@auxout{}{%
            \string\newlabel{sub@#1}%
            {{\@nameuse{@@thesub\@captype}}%
            {\thepage}}}%
        \LWR@write@lwarplabel{sub@#1}%          lwarp
        \@esphack
    }
}
```

§ 695.3 Page layout

memoir already set the page size to a default, so it must be forced large for lwarp's use, to avoid tag overflows off the page.

```
\setstocksize{190in}{20in}
\setlrmarginsandblock{2in}{2in}{*}
\setulmarginsandblock{1in}{1in}{*}
```

```
\renewcommand*{\stockavi}{}
\renewcommand*{\stockav}{}
\renewcommand*{\stockaiv}{}
\renewcommand*{\stockaiii}{}
```

```
\renewcommand*{\stockavii}{}
\renewcommand*{\stockbvi}{}
\renewcommand*{\stockbv}{}
\renewcommand*{\stockbiv}{}
\renewcommand*{\stockbiii}{}
\renewcommand*{\stockbvii}{}
% \renewcommand*{\stockmetriccrownvo}{}% in docs but not in the package
\renewcommand*{\stockmlargecrownvo}{}
\renewcommand*{\stockmdemyvo}{}
\renewcommand*{\stockmsmallroyalvo}{}
\renewcommand*{\pageavi}{}
\renewcommand*{\pageavii}{}
\renewcommand*{\pageav}{}
\renewcommand*{\pageaiv}{}
\renewcommand*{\pageaiii}{}
\renewcommand*{\pagebvi}{}
\renewcommand*{\pagebvii}{}
\renewcommand*{\pagebv}{}
\renewcommand*{\pagebiv}{}
\renewcommand*{\pagebiii}{}
% \renewcommand*{\pagemetriccrownvo}{}% in docs but not in the package
\renewcommand*{\pagemlargecrownvo}{}
\renewcommand*{\pagemdemyvo}{}
\renewcommand*{\pagemsmallroyalvo}{}

\renewcommand*{\stockdbill}{}
\renewcommand*{\stockstatement}{}
\renewcommand*{\stockexecutive}{}
\renewcommand*{\stockletter}{}
\renewcommand*{\stockold}{}
\renewcommand*{\stocklegal}{}
\renewcommand*{\stockledger}{}
\renewcommand*{\stockbroadsheet}{}
\renewcommand*{\pagedbill}{}
\renewcommand*{\pagestatement}{}
\renewcommand*{\pageexecutive}{}
\renewcommand*{\pageletter}{}
\renewcommand*{\pageold}{}
\renewcommand*{\pagelegal}{}
\renewcommand*{\pageledger}{}
\renewcommand*{\pagebroadsheet}{}

\renewcommand*{\stockpottvo}{}
\renewcommand*{\stockfoolscapvo}{}
\renewcommand*{\stockcrownvo}{}
\renewcommand*{\stockpostvo}{}
\renewcommand*{\stocklargecrownvo}{}
\renewcommand*{\stocklargepostvo}{}
\renewcommand*{\stocksmalldemyvo}{}
\renewcommand*{\stockdemyvo}{}
\renewcommand*{\stockmediumvo}{}
\renewcommand*{\stocksmallroyalvo}{}
\renewcommand*{\stockroyalvo}{}
\renewcommand*{\stocksuperroyalvo}{}
\renewcommand*{\stockimperialvo}{}
\renewcommand*{\pagepottvo}{}
\renewcommand*{\pagefoolscapvo}{}
\renewcommand*{\pagecrownvo}{}
\renewcommand*{\pagepostvo}{}
\renewcommand*{\pagelargecrownvo}{}
```

```
\renewcommand*{\pagelargepostvo}{}
\renewcommand*{\pagesmalldemyvo}{}
\renewcommand*{\pagedemyvo}{}
\renewcommand*{\pagemediumvo}{}
\renewcommand*{\pagesmallroyalvo}{}
\renewcommand*{\pageroyalvo}{}
\renewcommand*{\pagesuperroyalvo}{}
\renewcommand*{\pageimperialvo}{}

\renewcommand*{\memfontfamily}{}
\renewcommand*{\memfontenc}{}
\renewcommand*{\memfontpack}{}

\renewcommand*{\anyptfilebase}{}
\renewcommand*{\anyptsize}{10}

\renewcommand*{\setstocksize}[2]{}
\renewcommand*{\settrimmedsize}[3]{}
\renewcommand*{\settrims}[2]{}

% \newlength{\lxvchars}
% \setlength{\lxvchars}{305pt}
% \newlength{\xlvchars}
% \setlength{\xlvchars}{190pt}
\renewcommand*{\setxlvchars}[1]{}
\renewcommand*{\setlxvchars}[1]{}

\renewcommand*{\settypeblocksize}[3]{}
\renewcommand*{\setlrmargins}[3]{}
\renewcommand*{\setlrmarginsandblock}[3]{}
\renewcommand*{\setbinding}[1]{}
\renewcommand*{\setulmargins}[3]{}
\renewcommand*{\setulmarginsandblock}[3]{}
\renewcommand*{\setcolsepandrule}[2]{}

\renewcommand*{\setheadfoot}[2]{}
\renewcommand*{\setheaderspaces}[3]{}
\renewcommand*{\setmarginnotes}[3]{}
\renewcommand*{\setfootins}[2]{}
\renewcommand*{\checkandfixthelayout}[1][]{}
\renewcommand*{\checkthelayout}[1]{}
\renewcommand*{\fixthelayout}{}
%
% \newlength{\stockheight}
% \newlength{\trimtop}
% \newlength{\trimedge}
% \newlength{\stockwidth}
% \newlength{\spinemargin}
% \newlength{\foremargin}
% \newlength{\uppermargin}
% \newlength{\headmargin}
%
\renewcommand*{\typeoutlayout}{}
\renewcommand*{\typeoutstandardlayout}{}
\renewcommand*{\settypeoutlayoutunit}[1]{}
\renewcommand*{\fixpdflayout}{}
\renewcommand*{\fixdvipslayout}{}

\renewcommand*{\medievalpage}[1][]{}
\renewcommand*{\isopage}[1][]{}
```

```
\renewcommand*{\semiisopage}[1][]{}

\renewcommand{\setpagebl}[3]{}
\renewcommand{\setpageml}[3]{}
\renewcommand{\setpagetl}[3]{}
\renewcommand{\setpagetm}[3]{}
\renewcommand{\setpagetr}[3]{}
\renewcommand{\setpagemr}[3]{}
\renewcommand{\setpagebr}[3]{}
\renewcommand{\setpagebm}[3]{}
\renewcommand{\setpagecc}[3]{}
```

§ 695.4 Text and fonts

```
\let\miniscule\tiny
\let\HUGE\Huge

\renewcommand*{\abnormalparskip}[1]{}
\renewcommand*{\nonzeroparskip}{}
\renewcommand*{\traditionalparskip}{}

\let\onelineskip\baselineskip

\let\OnehalfSpacing\onehalfspacing
\let\DoubleSpacing\doublespacing
\renewcommand*{\setPagenoteSpacing}[1]{}
\renewcommand*{\setFloatSpacing}[1]{}

\renewcommand{\SingleSpacing}{\@ifstar\singlespacing\singlespacing}

\let\setSingleSpace\SetSinglespace
\let\SingleSpace\singlespace
\let\endSingleSpace\endsinglespace
\let\Spacing\spacing
\let\endSpacing\endspacing
\let\OnehalfSpace\onehalfspace
\let\endOnehalfSpace\endonehalfspace
\csletcs{OnehalfSpace*}{onehalfspace}
\csletcs{endOnehalfSpace*}{endonehalfspace}
\let\DoubleSpace\doublespace
\let\endDoubleSpace\enddoublespace
\csletcs{DoubleSpace*}{doublespace}
\csletcs{endDoubleSpace*}{enddoublespace}
\renewcommand*{\setDisplayskipStretch}[1]{}
\renewcommand*{\memdskipstretch}{}
\renewcommand*{\noDisplayskipStretch}{}
\renewcommand*{\memdskips}{}

\renewcommand*{\midsloppy}{}
\renewenvironment*{midsloppypar}{}{}

\renewcommand*{\sloppybottom}{}
```

§ 695.5 Titles

```
\csletcs{titlingpage*}{titlingpage}
\csletcs{endtitlingpage*}{endtitlingpage}
\let\titlingpageend\relax
\newcommand{\titlingpageend}[2]{}
\let\andnext\and
```

```
\renewcommand*{\thanksmarkstyle}[1]{}

\renewcommand{\thanksfootmark}{%
    \thanksscript{\tamark}%
}

% \newlength{\thanksmarksep}% already provided by memoir

\renewcommand\titlingpageend[2]{}
```

§ 695.6 Abstracts

```
% \newlength{\absindent}
% \newlength{\absparsep}
\renewcommand*{\abstractcol}{}
\renewcommand*{\abstractintoc}{}
\renewcommand*{\abstractnum}{}
\renewcommand*{\abstractrunin}{}
```

§ 695.7 Docment divisions

\book * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\book}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{book}%
}

\def\@apppage{%
    \part*{\appendixpagename}
}
\renewcommand\mempreaddapppagetotochook{}
\renewcommand\mempostaddapppagetotochook{}

\def\@sapppage{%
    \part*{\appendixpagename}
}


\DeclareDocumentCommand{\mainmatter}{s}{%
    \booltrue{LWR@mainmatter}%
}

\DeclareDocumentCommand{\frontmatter}{s}{%
    \boolfalse{LWR@mainmatter}%
}


\renewcommand*{\raggedbottomsection}{}
\renewcommand*{\normalbottomsection}{}
\renewcommand*{\bottomsectionskip}{}
\renewcommand*{\bottomsectionpenalty}{}
\csletcs{appendixpage*}{appendixpage}
\renewcommand*{\namedsubappendices}{}
\renewcommand*{\unnamedsubappendices}{}
\renewcommand*{\beforebookskip}{}
\renewcommand*{\afterbookskip}{}
\renewcommand*{\beforepartskip}{}
\renewcommand*{\afterpartskip}{}
\renewcommand*{\midbookskip}{}
```

```
\renewcommand*{\midpartskip}{}
\renewcommand*{\printbookname}{}
\renewcommand*{\booknamefont}{}
\renewcommand*{\booknamenum}{}
\renewcommand*{\printbooknum}{}
\renewcommand*{\booknumfont}{}
\renewcommand*{\printpartname}{}
\renewcommand*{\partnamefont}{}
\renewcommand*{\partnamenum}{}
\renewcommand*{\printpartnum}{}
\renewcommand*{\partnumfont}{}
\renewcommand*{\printbooktitle}[1]{}
\renewcommand*{\booktitlefont}{}
\renewcommand{\printparttitle}[1]{}
\renewcommand*{\parttitlefont}{}
\renewcommand*{\bookpageend}{}
\renewcommand*{\bookblankpage}{}
\renewcommand*{\nobookblankpage}{}
\renewcommand*{\partpageend}{}
\renewcommand*{\partblankpage}{}
\renewcommand*{\nopartblankpage}{}
\RenewDocumentCommand{\newleadpage}{s o m m}{}% todo
\RenewDocumentCommand{\renewleadpage}{s o m m}{}% todo
\renewcommand*{\leadpagetoclevel}{chapter}

\renewcommand*{\openright}{}
\renewcommand*{\openleft}{}
\renewcommand*{\openany}{}
\renewcommand*{\clearforchapter}{}
\renewcommand*{\memendofchapterhook}{}
\renewcommand*{\chapterheadstart}{}
% \newlength{\beforechapskip}
\renewcommand*{\afterchapternum}{}
% \newlength{\midchapskip}
\renewcommand*{\afterchaptertitle}{}
% \newlength{\afterchapskip}
\renewcommand*{\printchaptername}{}
\renewcommand*{\chapnamefont}{}
\renewcommand*{\chapternamenum}{}
\renewcommand*{\printchapternum}{}
\renewcommand*{\chapnumfont}{}
\renewcommand{\printchaptertitle}[1]{}
\renewcommand*{\chaptitlefont}{}
\renewcommand*{\printchapternonum}{}
\renewcommand*{\indentafterchapter}{}
\renewcommand*{\noindentafterchapter}{}
\renewcommand*{\insertchapterspace}{}

\renewcommand*{\chapterstyle}[1]{}
\renewcommand{\makechapterstyle}[2]{}
\renewcommand*{\chapindent}{}
\let\chapterprecis\cftchapterprecis
\let\chapterprecishere\cftchapterprecishere
\let\chapterprecistoc\cftchapterprecistoc
\renewcommand*{\precisfont}{}
\renewcommand*{\prechapterprecis}{}
\renewcommand*{\postchapterprecis}{}
\renewcommand{\precistoctext}[1]{}
\renewcommand*{\precistocfont}{}
\renewcommand*{\precistocformat}{}
```

```
% \newlength{\prechapterprecisshift}

\renewcommand*{\setbeforesecskip}[1]{}
\renewcommand*{\setaftersecskip}[1]{}
\renewcommand*{\setsecindent}[1]{}
\renewcommand*{\setsecheadstyle}[1]{}
\renewcommand*{\setbeforesubsecskip}[1]{}
\renewcommand*{\setaftersubsecskip}[1]{}
\renewcommand*{\setsubsecindent}[1]{}
\renewcommand*{\setsubsecheadstyle}[1]{}
\renewcommand*{\setbeforesubsubsecskip}[1]{}
\renewcommand*{\setaftersubsubsecskip}[1]{}
\renewcommand*{\setsubsubsecindent}[1]{}
\renewcommand*{\setsubsubsecheadstyle}[1]{}
\renewcommand*{\setbeforeparaskip}[1]{}
\renewcommand*{\setafterparaskip}[1]{}
\renewcommand*{\setparaindent}[1]{}
\renewcommand*{\setparaheadstyle}[1]{}
\renewcommand*{\setbeforesubparaskip}[1]{}
\renewcommand*{\setaftersubparaskip}[1]{}
\renewcommand*{\setsubparaindent}[1]{}
\renewcommand*{\setsubparaheadstyle}[1]{}
\renewcommand{\@hangfrom}[1]{#1}
\renewcommand{\sethangfrom}[1]{}
\renewcommand{\setsecnumformat}[1]{}

\renewcommand*{\hangsecnum}{}
\renewcommand*{\defaultsecnum}{}

\renewcommand*{\sechook}{}
\renewcommand{\setsechook}[1]{}
\renewcommand*{\subsechook}{}
\renewcommand{\setsubsechook}[1]{}
\renewcommand*{\subsubsechook}{}
\renewcommand{\setsubsubsechook}[1]{}
\renewcommand*{\parahook}{}
\renewcommand{\setparahook}[1]{}
\renewcommand*{\subparahook}{}
\renewcommand{\setsubparahook}[1]{}

\RenewDocumentCommand{\plainbreak}{s m}{\begin{center}~\end{center}}

\RenewDocumentCommand{\fancybreak}{s +m}{%
    \begin{center}#2\end{center}%
}

\RenewDocumentCommand{\plainfancybreak}{s m m +m}{%
    \begin{center}#4\end{center}%
}

\RenewDocumentCommand{\pfbreak}{s}{%
    \begin{center}
    \pfbreakdisplay
    \end{center}
}

% \newlength{\pfbreakskip}
\renewcommand{\pfbreakdisplay}{*\quad*\quad*}

\renewcommand{\makeheadstyles}[2]{}
```

```
\renewcommand*{\headstyles}[1]{}
```

§ 695.8 Pagination and headers

```
\renewcommand*{\savepagenumber}{}
\renewcommand*{\restorepagenumber}{}
\renewcommand*{\uppercaseheads}{}
\renewcommand*{\nouppercaseheads}{}

\renewcommand*{\bookpagemark}[1]{}
\renewcommand*{\partmark}[1]{}
\renewcommand*{\bibmark}{}
\renewcommand*{\indexmark}{}
\renewcommand*{\glossarymark}{}

\LWR@origpagestyle{empty}
\renewcommand*{\ps@empty}{}
\renewcommand*{\makepagestyle}[1]{}
\renewcommand*{\emptypshook}{}%
% \renewcommand*{\empty@oddhead}{}
% \renewcommand*{\empty@oddfoot}{}
% \renewcommand*{\empty@evenhead}{}
% \renewcommand*{\empty@evenfoot}{}
\renewcommand*{\@oddhead}{}
\renewcommand*{\@oddfoot}{}
\renewcommand*{\@evenhead}{}
\renewcommand*{\@evenfoot}{}
\renewcommand*{\aliaspagestyle}[2]{}
\renewcommand*{\copypagestyle}[2]{}

\renewcommand*{\makeevenhead}[4]{}
\renewcommand*{\makeoddhead}[4]{}
\renewcommand*{\makeevenfoot}[4]{}
\renewcommand*{\makeoddfoot}[4]{}
\renewcommand*{\makerunningwidth}[3]{}
% \newlength{\headwidth}
\renewcommand*{\makeheadrule}[3]{}
\renewcommand*{\makefootrule}[3]{}
\renewcommand*{\makeheadfootruleprefix}[3]{}
% \newlength{\normalrulethickness}
% \setlength{\normalrulethickness}{.4pt}
% \newlength{\footruleheight}
% \newlength{\footruleskip}
\renewcommand*{\makeheadposition}[5]{}
\renewcommand{\makepsmarks}[2]{}
\renewcommand*{\makeheadfootstrut}[3]{}


\renewcommand{\createmark}[5]{\csdef{#1mark}[1]{}}
\renewcommand{\createplainmark}[3]{\csdef{#1mark}{}}

\renewcommand{\memUChead}[1]{}
\renewcommand*{\clearplainmark}[1]{}
\renewcommand*{\clearmark}[1]{}
\renewcommand{\addtopsmarks}[3]{}
\renewcommand{\ifonlyfloats}[2]{#2}
\renewcommand*{\mergepagefloatstyle}[3]{}

\renewcommand*{\framepichead}{}
\renewcommand*{\framepictextfoot}{}
\renewcommand*{\framepichook}{}
```

```
\renewcommand*{\showheadfootlocoff}{}
\renewcommand*{\showtextblocklocoff}{}
```

§ 695.9 Paragraphs and lists

```
\renewcommand{\hangfrom}[1]{#1}
\let\centerfloat\centering
\renewcommand*{\raggedyright}[1][]{}
% \newlength{\ragrparindent}
\renewcommand{\sourceatright}[2][]{\attribution{#2}}
\let\memorigdbs\LWR@endofline

\renewcommand*{\memorigpar}{\par}

\let\atcentercr\LWR@endofline

\renewcommand*{\linenottooshort}[1][]{}
\renewcommand*{\russianpar}{}
\renewcommand*{\lastlinerulefill}{}
\renewcommand*{\lastlineparrule}{}
\renewcommand*{\justlastraggedleft}{}
\renewcommand*{\raggedrightthenleft}{}
\renewcommand*{\leftcenterright}{}

\renewcommand{\leftspringright}[4]{%
    \begin{minipage}{#1\linewidth}#3\end{minipage}\qquad%
   \begin{minipage}{#2\linewidth}\begin{flushright}#4\end{flushright}\end{minipage}%
}

\renewenvironment*{blockdescription}
{\LWR@descriptionstart\LWR@origdescription}
{\enddescription}

\renewcommand*{\blockdescriptionlabel}[1]{\textbf{#1}}
\renewenvironment*{labelled}[1]{\begin{description}}{\end{description}}
\renewenvironment*{flexlabelled}[6]{\begin{description}}{\end{description}}
\renewcommand*{\tightlists}{}
\renewcommand*{\defaultlists}{}
\RenewDocumentCommand{\firmlists}{s}{}
\renewcommand*{\firmlist}{}
\renewcommand*{\tightlist}{}
\renewcommand*{\zerotrivseps}{}
\renewcommand*{\savetrivseps}{}
\renewcommand*{\restoretrivseps}{}
```

§ 695.10 Contents lists

```
\csletcs{tableofcontents*}{tableofcontents}
\csletcs{listoffigures*}{listoffigures}
\csletcs{listoftables*}{listoftables}
\renewenvironment{KeepFromToc}{}{}
\renewcommand*{\onecoltocetc}{}
\renewcommand*{\twocoltocetc}{}
\renewcommand*{\ensureonecol}{}
\renewcommand*{\restorefromonecol}{}
\renewcommand*{\doccoltocetc}{}

\renewcommand{\tocheadstart}{}
\renewcommand{\printtoctitle}[1]{}
\renewcommand{\tocmark}{}
```

```
\renewcommand{\aftertoctitle}{}
\renewcommand{\lofheadstart}{}
\renewcommand{\printloftitle}[1]{}
\renewcommand{\lofmark}{}
\renewcommand{\afterloftitle}{}
\renewcommand{\lotheadstart}{}
\renewcommand{\printlottitle}[1]{}
\renewcommand{\lotmark}{}
\renewcommand{\afterlottitle}{}

\renewcommand*{\setpnumwidth}[1]{}
\renewcommand*{\setrmarg}[1]{}
\renewcommand*{\cftbookbreak}{}
\renewcommand*{\cftpartbreak}{}
\renewcommand*{\cftchapterbreak}{}

% \newlength{\cftbeforebookskip}
% \newlength{\cftbookindent}
% \newlength{\cftbooknumwidth}
\renewcommand*{\cftbookfont}{}
\renewcommand*{\cftbookname}{}
\renewcommand*{\cftbookpresnum}{}
\renewcommand*{\cftbookaftersnum}{}
\renewcommand*{\cftbookaftersnumb}{}
\renewcommand*{\cftbookleader}{}
\renewcommand*{\cftbookdotsep}{1}
\renewcommand*{\cftbookpagefont}{}
\renewcommand*{\cftbookafterpnum}{}
\renewcommand*{\cftbookformatpnum}[1]{}
\renewcommand*{\cftbookformatpnumhook}[1]{}
```

Part is already defined by tocloft.

```
% \newlength{\cftbeforechapterskip}
% \newlength{\cftchapterindent}
% \newlength{\cftchapternumwidth}
\renewcommand*{\cftchapterfont}{}
\renewcommand*{\cftchaptername}{}
\renewcommand*{\cftchapterpresnum}{}
\renewcommand*{\cftchapteraftersnum}{}
\renewcommand*{\cftchapteraftersnumb}{}
\renewcommand*{\cftchapterleader}{}
\renewcommand*{\cftchapterdotsep}{1}
\renewcommand*{\cftchapterpagefont}{}
\renewcommand*{\cftchapterafterpnum}{}
\renewcommand*{\cftchapterformatpnum}[1]{}
\renewcommand*{\cftchapterformatpnumhook}[1]{}

% \newlength{\cftbeforesectionskip}
% \newlength{\cftsectionindent}
% \newlength{\cftsectionnumwidth}
\renewcommand*{\cftsectionfont}{}
\renewcommand*{\cftsectionname}{}
\renewcommand*{\cftsectionpresnum}{}
\renewcommand*{\cftsectionaftersnum}{}
\renewcommand*{\cftsectionaftersnumb}{}
\renewcommand*{\cftsectionleader}{}
\renewcommand*{\cftsectiondotsep}{1}
\renewcommand*{\cftsectionpagefont}{}
\renewcommand*{\cftsectionafterpnum}{}
```

```
\renewcommand*{\cftsectionformatpnum}[1]{}
\renewcommand*{\cftsectionformatpnumhook}[1]{}

% \newlength{\cftbeforesubsectionskip}
% \newlength{\cftsubsectionindent}
% \newlength{\cftsubsectionnumwidth}
\renewcommand*{\cftsubsectionfont}{}
\renewcommand*{\cftsubsectionname}{}
\renewcommand*{\cftsubsectionpresnum}{}
\renewcommand*{\cftsubsectionaftersnum}{}
\renewcommand*{\cftsubsectionaftersnumb}{}
\renewcommand*{\cftsubsectionleader}{}
\renewcommand*{\cftsubsectiondotsep}{1}
\renewcommand*{\cftsubsectionpagefont}{}
\renewcommand*{\cftsubsectionafterpnum}{}
\renewcommand*{\cftsubsectionformatpnum}[1]{}
\renewcommand*{\cftsubsectionformatpnumhook}[1]{}

% \newlength{\cftbeforesubsubsectionskip}
% \newlength{\cftsubsubsectionindent}
% \newlength{\cftsubsubsectionnumwidth}
\renewcommand*{\cftsubsubsectionfont}{}
\renewcommand*{\cftsubsubsectionname}{}
\renewcommand*{\cftsubsubsectionpresnum}{}
\renewcommand*{\cftsubsubsectionaftersnum}{}
\renewcommand*{\cftsubsubsectionaftersnumb}{}
\renewcommand*{\cftsubsubsectionleader}{}
\renewcommand*{\cftsubsubsectiondotsep}{1}
\renewcommand*{\cftsubsubsectionpagefont}{}
\renewcommand*{\cftsubsubsectionafterpnum}{}
\renewcommand*{\cftsubsubsectionformatpnum}[1]{}
\renewcommand*{\cftsubsubsectionformatpnumhook}[1]{}

% \newlength{\cftbeforeparagraphskip}
% \newlength{\cftparagraphindent}
% \newlength{\cftparagraphnumwidth}
\renewcommand*{\cftparagraphfont}{}
\renewcommand*{\cftparagraphname}{}
\renewcommand*{\cftparagraphpresnum}{}
\renewcommand*{\cftparagraphaftersnum}{}
\renewcommand*{\cftparagraphaftersnumb}{}
\renewcommand*{\cftparagraphleader}{}
\renewcommand*{\cftparagraphdotsep}{1}
\renewcommand*{\cftparagraphpagefont}{}
\renewcommand*{\cftparagraphafterpnum}{}
\renewcommand*{\cftparagraphformatpnum}[1]{}
\renewcommand*{\cftparagraphformatpnumhook}[1]{}

% \newlength{\cftbeforesubparagraphskip}
% \newlength{\cftsubparagraphindent}
% \newlength{\cftsubparagraphnumwidth}
\renewcommand*{\cftsubparagraphfont}{}
\renewcommand*{\cftsubparagraphname}{}
\renewcommand*{\cftsubparagraphpresnum}{}
\renewcommand*{\cftsubparagraphaftersnum}{}
\renewcommand*{\cftsubparagraphaftersnumb}{}
\renewcommand*{\cftsubparagraphleader}{}
\renewcommand*{\cftsubparagraphdotsep}{1}
\renewcommand*{\cftsubparagraphpagefont}{}
\renewcommand*{\cftsubparagraphafterpnum}{}
\renewcommand*{\cftsubparagraphformatpnum}[1]{}
```

```
\renewcommand*{\cftsubparagraphformatpnumhook}[1]{}

% \newlength{\cftbeforefigureskip}
% \newlength{\cftfigureindent}
% \newlength{\cftfigurenumwidth}
\renewcommand*{\cftfigurefont}{}
\renewcommand*{\cftfigurename}{}
\renewcommand*{\cftfigurepresnum}{}
\renewcommand*{\cftfigureaftersnum}{}
\renewcommand*{\cftfigureaftersnumb}{}
\renewcommand*{\cftfigureleader}{}
\renewcommand*{\cftfiguredotsep}{1}
\renewcommand*{\cftfigurepagefont}{}
\renewcommand*{\cftfigureafterpnum}{}
\renewcommand*{\cftfigureformatpnum}[1]{}
\renewcommand*{\cftfigureformatpnumhook}[1]{}

% \newlength{\cftbeforesubfigureskip}
% \newlength{\cftsubfigureindent}
% \newlength{\cftsubfigurenumwidth}
\newcommand*{\cftsubfigurefont}{}
\newcommand*{\cftsubfigurename}{}
\newcommand*{\cftsubfigurepresnum}{}
\newcommand*{\cftsubfigureaftersnum}{}
\newcommand*{\cftsubfigureaftersnumb}{}
\newcommand*{\cftsubfigureleader}{}
\newcommand*{\cftsubfiguredotsep}{1}
\newcommand*{\cftsubfigurepagefont}{}
\newcommand*{\cftsubfigureafterpnum}{}
\newcommand*{\cftsubfigureformatpnum}[1]{}
\newcommand*{\cftsubfigureformatpnumhook}[1]{}

% \newlength{\cftbeforetableskip}
% \newlength{\cfttableindent}
% \newlength{\cfttablenumwidth}
\renewcommand*{\cfttablefont}{}
\renewcommand*{\cfttablename}{}
\renewcommand*{\cfttablepresnum}{}
\renewcommand*{\cfttableaftersnum}{}
\renewcommand*{\cfttableaftersnumb}{}
\renewcommand*{\cfttableleader}{}
\renewcommand*{\cfttabledotsep}{1}
\renewcommand*{\cfttablepagefont}{}
\renewcommand*{\cfttableafterpnum}{}
\renewcommand*{\cfttableformatpnum}[1]{}
\renewcommand*{\cfttableformatpnumhook}[1]{}

% \newlength{\cftbeforesubtableskip}
% \newlength{\cftsubtableindent}
% \newlength{\cftsubtablenumwidth}
\newcommand*{\cftsubtablefont}{}
\newcommand*{\cftsubtablename}{}
\newcommand*{\cftsubtablepresnum}{}
\newcommand*{\cftsubtableaftersnum}{}
\newcommand*{\cftsubtableaftersnumb}{}
\newcommand*{\cftsubtableleader}{}
\newcommand*{\cftsubtabledotsep}{1}
\newcommand*{\cftsubtablepagefont}{}
\newcommand*{\cftsubtableafterpnum}{}
\newcommand*{\cftsubtableformatpnum}[1]{}
\newcommand*{\cftsubtableformatpnumhook}[1]{}
```

```
\renewcommand*{\booknumberline}[1]{}
\renewcommand*{\partnumberline}[1]{}
\renewcommand*{\chapternumberline}[1]{}
\renewcommand*{\numberlinehook}[1]{}
% \renewcommand*{\cftwhatismyname}{}%
\renewcommand*{\booknumberlinehook}[1]{}
\renewcommand*{\partnumberlinehook}[1]{}
\renewcommand*{\chapternumberlinehook}[1]{}
\renewcommand{\numberlinebox}[2]{}
\renewcommand{\booknumberlinebox}[2]{}
\renewcommand{\partnumberlinebox}[2]{}
\renewcommand{\chapternumberlinebox}[2]{}
%
% \newlength{\cftparfillskip}
\renewcommand*{\cftpagenumbersoff}[1]{}
\renewcommand*{\cftpagenumberson}[1]{}
\renewcommand*{\cftlocalchange}[3]{}
\renewcommand*{\cftaddtitleline}[4]{}
\renewcommand*{\cftaddnumtitleline}[4]{}
\renewcommand{\cftinsertcode}[2]{}
\renewcommand{\cftinserthook}[2]{}
\renewcommand{\settocpreprocessor}[2]{}
\DeclareRobustCommand{\cftpagenumbersoff}[1]{}
\DeclareRobustCommand{\cftpagenumberson}[1]{}
```

§ 695.11 Floats and captions

\@xfloat

\@xdblfloat

Reestablish lwarp's takeover the float handing, which memoir tried to grab:

```
\AtBeginDocument{
\def\@xfloat #1[#2]{%
    \LWR@floatbegin{#1}[#2]
    \normalsize
    \@nameuse{#1adjustment}%
    \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
}
\def\@xdblfloat #1[#2]{%
    \LWR@floatbegin{#1}[#2]
    \normalsize
    \@nameuse{#1adjustment}%
    \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
}
}
```

\newfloat

[⟨*1: within*⟩] {⟨*2: type*⟩} {⟨*3: ext*⟩} {⟨*4: capname*⟩}

```
\RenewDocumentCommand{\newfloat}{o m m m}{%
    \def\LWR@tempone{#4}%
    \def\LWR@temptwo{\@nameuse{#2name}}%
   \ifdefequal{\LWR@tempone}{\LWR@temptwo}{% recursive name, already defined
        \IfValueTF{#1}%
            {\DeclareFloatingEnvironment[fileext=#3,within=#1]{#2}}%
            {\DeclareFloatingEnvironment[fileext=#3]{#2}}%
    }{% not recursive name
        \IfValueTF{#1}%
         {\DeclareFloatingEnvironment[fileext=#3,within=#1,name={#4}]{#2}}%
            {\DeclareFloatingEnvironment[fileext=#3,name={#4}]{#2}}%
    }%
```

newfloat package automatically creates the \listof command for new floats, but float does not, so remove \listof here in case it is manually created later.

```
    \cslet{listof#2s}\relax%
    \cslet{listof#2es}\relax%
}
```

\newlistof [⟨*within*⟩] {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the \newfloat mechanism. Note that memoir uses a different syntax than tocloft for the name.

```
\RenewDocumentCommand{\newlistof}{o m m m}
{%
    \IfValueTF{#1}%
        {\newlistentry[#1]{#2}{#3}{0}}%
        {\newlistentry{#2}{#3}{0}}%
    \@namedef{ext@#2}{#3}%
    \@ifundefined{c@#3depth}{\newcounter{#3depth}}{}%
    \setcounter{#3depth}{1}%
    \@namedef{#3mark}{}%
    \@namedef{#2}{\LWR@listof{#2}{#4}}%
    \@namedef{@cftmake#3title}{}%
    \@ifundefined{cftbefore#3titleskip}{%
        \expandafter\newlength\csname cftbefore#3titleskip\endcsname%
        \expandafter\newlength\csname cftafter#3titleskip\endcsname%
    }{}%
    \@namedef{cft#3titlefont}{}%
    \@namedef{cftafter#3title}{}%
    \@namedef{cft#3prehook}{}%
    \@namedef{cft#3posthook}{}%
}

\renewcommand{\setfloatadjustment}[2]{}
```

Borrowed from the lwarp version of keyfloat:

```
\NewDocumentEnvironment{KFLTmemoir@marginfloat}{O{-1.2ex} m}
{% start
   \LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}(note){marginblock}%
    \renewcommand*{\@captype}{#2}%
}
{%
    \endLWR@BlockClassWP%
}

\DeclareDocumentEnvironment{marginfigure}{o}
  {\begin{KFLTmemoir@marginfloat}{figure}}
  {\end{KFLTmemoir@marginfloat}}

\DeclareDocumentEnvironment{margintable}{o}
  {\begin{KFLTmemoir@marginfloat}{table}}
  {\end{KFLTmemoir@marginfloat}}

\renewcommand{\setmarginfloatcaptionadjustment}[2]{}
\renewcommand{\setmpjustification}[2]{}
\renewcommand*{\mpjustification}{}
\renewcommand*{\setfloatlocations}[2]{}
\DeclareDocumentCommand{\suppressfloats}{o}{}
\renewcommand*{\FloatBlock}{}
```

```
\renewcommand*{\FloatBlockAllowAbove}{}
\renewcommand*{\FloatBlockAllowBelow}{}
\renewcommand*{\setFloatBlockFor}{}

\renewcommand{\captiontitlefinal}[1]{}
```

`\flegtable`, `\flegfigure`, `\flegtoctable`, `\flegtocfigure` are defined by memoir using `\newfloat`. These are defined with an `@` in ccaption.

```
\renewcommand{\flegtable}{\tablename}
\renewcommand{\flegfigure}{\figurename}
\renewcommand{\flegtoctable}{}
\renewcommand{\flegtocfigure}{}


\renewcommand{\@makesubfloatcaption}[2]{%
    \minipagefullwidth
    \begin{minipage}{\linewidth}%
    #1 \ignorespaces #2 \unskip%
    \end{minipage}
}

\renewcommand*{\tightsubcaptions}{}
\renewcommand*{\loosesubcaptions}{}

\renewcommand*{\subcaptionsize}[1]{}
\renewcommand*{\subcaptionlabelfont}[1]{}
\renewcommand*{\subcaptionfont}[1]{}
\renewcommand*{\subcaptionstyle}[1]{}

\renewcommand*{\hangsubcaption}{}
\renewcommand*{\shortsubcaption}{}
\renewcommand*{\normalsubcaption}{}

\RenewDocumentEnvironment{sidecaption}{o m o}
{}
{%
    \IfValueTF{#1}{\caption[#1]{#2}}{\caption{#2}}%
    \IfValueT{#3}{\label{#3}}%
}

% \newlength{\sidecapwidth}
% \newlength{\sidecapsep}
\renewcommand*{\setsidecaps}[2]{}
\renewcommand*{\sidecapmargin}[1]{}
% \newif\ifscapmargleft
\scapmargleftfalse
\renewcommand*{\setsidecappos}[1]{}
```

Env `sidecontcaption`

```
\RenewDocumentEnvironment{sidecontcaption}{m o}
{}
{%
    \ifdef{\ContinuedFloat}%
        {\ContinuedFloat}%
        {\addtocounter{\@captype}{-1}}%
    \caption{#1}%
```

Without \@captype, the section is referred to instead.

```
    \IfValueT{#2}{\label[\@captype]{#2}}%
}
```

\sidenamedlegend does not appear to use the TOC argument.

```
\renewenvironment{sidenamedlegend}[2][]{
    \begin{center}
    \@nameuse{\@captype name}\CaptionSeparator#2
    \end{center}
}
{}

\renewenvironment{sidelegend}[1]
{\begin{center}
    #1

}
{\end{center}}

\renewcommand*{\sidecapstyle}{}
\renewcommand*{\overridescapmargin}[1]{}
% \newlength{\sidecapraise}
\renewcommand*{\sidecapfloatwidth}{\linewidth}

\LetLtxMacro\ctabular\tabular
\LetLtxMacro\endctabular\endtabular

\renewcommand{\autorows}[5][]{%
    #5%
}

\renewcommand{\autocols}[5][]{%
    #5%
}
```

§ 695.12 Footnotes and page notes

```
\renewcommand*{\feetabovefloat}{}
\renewcommand*{\feetbelowfloat}{}
\renewcommand*{\feetatbottom}{}

\renewcommand*{\verbfootnote}[2][]{%
    \PackageError{lwarp,memoir}%
    {Verbatim footnotes are not yet supported by lwarp}%
    {This may be improved some day.}%
}

\renewcommand*{\plainfootnotes}{}
\renewcommand*{\twocolumnfootnotes}{}
\renewcommand*{\threecolumnfootnotes}{}
\renewcommand*{\paragraphfootnotes}{}
\renewcommand*{\footfudgefiddle}{}

\renewcommand*{\newfootnoteseries}[1]{%
    \PackageError{lwarp,memoir}%
    {Memoir footnote series are not yet supported by lwarp}%
    {This may be improved some day.}%
}

```

```
\renewcommand*{\plainfootstyle}[1]{}
\renewcommand*{\twocolumnfootstyle}[1]{}
\renewcommand*{\threecolumnfootstyle}[1]{}
\renewcommand*{\paragraphfootstyle}[1]{}

\renewcommand*{\footfootmark}{}
\renewcommand*{\footmarkstyle}[1]{}

% \newlength{\footmarkwidth}
% \newlength{\footmarksep}
% \newlength{\footparindent}

\renewcommand*{\foottextfont}{}

\renewcommand*{\marginparmargin}[1]{}
\renewcommand*{\sideparmargin}[1]{}

\LetLtxMacro\sidepar\marginpar
\renewcommand*{\sideparfont}{}
\renewcommand*{\sideparform}{}
\LWR@providelength{\sideparvshift}

\renewcommand*{\parnopar}{}

\renewcommand{\sidebar}[1]{\begin{quote}#1\end{quote}}
\renewcommand*{\sidebarmargin}[1]{}
\renewcommand*{\sidebarfont}{}
\renewcommand*{\sidebarform}{}
% \newlength{\sidebarhsep}
% \newlength{\sidebarvsep}
% \newlength{\sidebarwidth}
% \newlength{\sidebartopsep}
\renewcommand{\setsidebarheight}[1]{}
\renewcommand*{\setsidebars}[6]{}
\renewcommand*{\footnotesatfoot}{}
\renewcommand*{\footnotesinmargin}{}

\LetLtxMacro\sidefootnote\footnote
\LetLtxMacro\sidefootnotemark\footnotemark
\LetLtxMacro\sidefootnotetext\footnotetext

\renewcommand*{\sidefootmargin}[1]{}
% \newlength{\sidefoothsep}
% \newlength{\sidefootvsep}
% \newlength{\sidefootwidth}
% \newlength{\sidefootadjust}
% \newlength{\sidefootheight}
\renewcommand*{\setsidefootheight}[1]{}
% \renewcommand*{\sidefootfont}{}% in docs but not in the package
\renewcommand*{\setsidefeet}[6]{}
\renewcommand*{\sidefootmarkstyle}[1]{}
\renewcommand*{\sidefoottextfont}{}
\renewcommand*{\sidefootform}{}


\renewcommand*{\continuousnotenums}{\pncontopttrue}% from pagenote
\renewcommand*{\notepageref}{}
\renewcommand*{\prenotetext}{}
\renewcommand*{\postnotetext}{}
\LetLtxMacro\printpageinnoteshyperref\printpageinnotes
\renewcommand*{\foottopagenote}{}
```

```
\renewcommand*{\pagetofootnote}{}
```

\m@m@wrpnote

\startnoteentrystart

To have cleveref work with page note labels, the following patch writes \thepagenote and also adds \arabic{pagenote} to the first argument written to the .ent file:

```
\startnoteentry{{\thepagenote}{\arabic{pagenote}}} . . .
```

The arabic value is required for cleveref. \thepagenote becomes \@firstoftwo#1 and the arabic value becomes \@secondoftwo#1.

⚠ \nameref Note that for print mode,\nameref print the section name where the page notes are declared in the text, but for HTML it prints the name where the page notes are printed.

```
\xpatchcmd{\m@m@wrpnote}
    {\string\startnoteentry{\thepagenote}}
    {\string\startnoteentry{{\thepagenote}{\arabic{pagenote}}}}
    {}
    {\LWR@patcherror{memoir}{m@m@wrpnote}}

\renewcommand\startnoteentrystart[4]{%
  \prenoteinnotes%
  \noteidinnotes{\@firstoftwo#1}{#2}%
  \@ifmtarg{#2}{%
%         \phantomsection\def\@currentlabel{#1}%            original
        \def\@currentlabel{\@firstoftwo#1}%                 lwarp
        \def\cref@currentlabel{%                            lwarp
            [pagenote][\@secondoftwo#1][]\@firstoftwo#1%    lwarp
        }%                                                  lwarp
  }{}%
  \pagenoteanchor{#4}%
  \pageinnotes{#3}%
  \prenotetext%
}
```

§ 695.13 Decorative text

```
\renewcommand*{\epigraphposition}[1]{}
\renewcommand*{\epigraphtextposition}[1]{}
\renewcommand*{\epigraphsourceposition}[1]{}
\renewcommand*{\epigraphfontsize}[1]{}
\renewcommand*{\epigraphforheader}[2][]{}
\renewcommand*{\epigraphpicture}{}
```

§ 695.14 Poetry

```
\renewcommand*{\vinphantom}{}
\renewcommand*{\vleftofline}[1]{#1}
% \let\linenumberfrequency\poemlines
% \renewcommand*{\linenumberfont}[1]{}

\DeclareDocumentCommand{\PoemTitle}{s o o m}{%
    \IfValueTF{#2}%
        {\poemtitle[#2]{#4}}%
        {\poemtitle{#4}}%
}

\renewcommand*{\NumberPoemTitle}{}
\renewcommand*{\PlainPoemTitle}{}
\renewcommand*{\poemtitlepstyle}{}
```

```
\renewcommand*{\poemtitlestarmark}[1]{}
\renewcommand*{\poemtitlestarpstyle}{}
\renewcommand*{\PoemTitleheadstart}{}
\renewcommand*{\printPoemTitlenonum}{}
\renewcommand*{\printPoemTitlenum}{}
\renewcommand*{\afterPoemTitlenum}{}
\renewcommand*{\printPoemTitletitle}[1]{}
\renewcommand*{\afterPoemTitle}{}
\newlength{\midpoemtitleskip}
\renewcommand*{\PoemTitlenumfont}{}
\renewcommand*{\PoemTitlefont}{}
```

§ 695.15 Boxes, verbatims and files

```
\renewenvironment{qframe}{\framed}{\endframed}
\renewenvironment{qshade}{\shaded}{\endshaded}
```

```
\renewcommand*{\setverbatimfont}[1]{}
\renewcommand*{\tabson}[1]{}
\renewcommand*{\tabsoff}{}
\renewcommand*{\wrappingon}{}
\renewcommand*{\wrappingoff}{}
\renewcommand*{\verbatimindent}{}
\renewcommand*{\verbatimbreakchar}[1]{}
```

```
\DefineVerbatimEnvironment{fboxverbatim}{Verbatim}{frame=single}
```

boxedverbatim is already defined by moreverb. boxedverbatim* does not appear to work at all, even in a minimal print memoir document.

```
\renewcommand*{\bvbox}{}
\renewcommand*{\bvtopandtail}{}
\renewcommand*{\bvsides}{}
\renewcommand*{\nobvbox}{}
% \newlength\bvboxsep
\renewcommand*{\bvtoprulehook}{}
\renewcommand*{\bvtopmidhook}{}
\renewcommand*{\bvendrulehook}{}
\renewcommand*{\bvleftsidehook}{}
\renewcommand*{\bvrightsidehook}{}
\renewcommand*{\bvperpagetrue}{}
\renewcommand*{\bvperpagefalse}{}
\renewcommand{\bvtopofpage}[1]{}
\renewcommand{\bvendofpage}[1]{}
\renewcommand*{\linenumberfrequency}[1]{}
\renewcommand*{\resetbvlinenumber}{}
\renewcommand*{\setbvlinenums}[2]{}
\renewcommand*{\linenumberfont}[1]{}
\renewcommand*{\bvnumbersinside}{}
\renewcommand*{\bvnumbersoutside}{}
```

§ 695.16 Cross referencing

```
\renewcommand*{\fref}[1]{\cref{#1}}
\renewcommand*{\tref}[1]{\cref{#1}}
\renewcommand*{\pref}[1]{\cpageref{#1}}
\renewcommand*{\Aref}[1]{\cref{#1}}
\renewcommand*{\Bref}[1]{\cref{#1}}
\renewcommand*{\Pref}[1]{\cref{#1}}
```

```
\renewcommand*{\Sref}[1]{\cref{#1}}
\renewcommand*{\figurerefname}{Figure}
\renewcommand*{\tablerefname}{Table}
\renewcommand*{\pagerefname}{page}
\renewcommand*{\bookrefname}{Book~}
\renewcommand*{\partrefname}{Part~}
\renewcommand*{\chapterrefname}{Chapter~}
\renewcommand*{\sectionrefname}{\S}
\renewcommand*{\appendixrefname}{Appendix~}
\LetLtxMacro\titleref\nameref
\renewcommand*{\headnameref}{}
\renewcommand*{\tocnameref}{}

\providecounter{LWR@currenttitle}

\renewcommand*{\currenttitle}{%
    \addtocounter{LWR@currenttitle}{1}%
    \label{currenttitle\arabic{LWR@currenttitle}}%
    \nameref{currenttitle\arabic{LWR@currenttitle}}%
}

\renewcommand*{\theTitleReference}[2]{}
\renewcommand*{\namerefon}{}
\renewcommand*{\namerefoff}{}
```

§ 695.17 Back matter

\@@wrindexhyp

Redefined to write the LWR@autoindex counter instead of page. Note that memoir has two versions, depending on the use of hyperref.

```
\AtBeginDocument{

\def\@@wrindexhyp#1||\\{%
    \addtocounter{LWR@autoindex}{1}%            lwarp
%     \ifshowindexmark\@showidx{#1}\fi
    \protected@write\@auxout{}%
%         {\string\@@wrindexm@m{\@idxfile}{#1}{\thepage}}%
        {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}% lwarp
```

The label is assigned after the file write to avoid conflict with cleveref.

```
    \label{LWRindex-\arabic{LWR@autoindex}}%    lwarp
    \endgroup
    \@esphack}%
```

\@@wrspindexhyp

\specialindex behaves like a regular \index, pointing to where \specialindex is used. If \specialindex is used inside a figure or table after the \caption, then the hyperlink will be given the name of that particular figure or table.

```
\def\@@wrspindexhyp#1||\\{%
    \addtocounter{LWR@autoindex}{1}%
%     \ifshowindexmark\@showidx{#1}\fi
    \protected@write\@auxout{}%
%         {\string\@@wrindexm@m{\@idxfile}{#1}{\@nameuse{the\@sptheidx}}}%
        {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
    \label{LWRindex-\arabic{LWR@autoindex}}%
    \endgroup
    \@esphack}%

```

```
}% \AtBeginDocument
```

\@spindex

Patched to append _html to the file:

```
\renewcommand{\@spindex}[2]{%
  \@ifundefined{#1@idxfile}%
  {\ifreportnoidxfile
     \@memwarn{Undefined index file #1}%
    \fi
    \begingroup
    \@sanitize
    \@nowrindex}%
  {\def\@idxfile{#1_html}%
   \def\@sptheidx{#2}%
   \begingroup
   \@sanitize
   \@wrspindex}}
```

\makeindex

Patched to use _html filename and \BaseJobname:

```
\catcode`\_=12%
\renewcommand*{\makeindex}[1][\BaseJobname]{%
  \if@filesw
    \def\gindex{\@bsphack%
      \@ifnextchar [{\@index}{\@index[\BaseJobname]}}
    \def\specialindex{\@bsphack\@spindex}%
    \makememindexhook
    \expandafter\newwrite\csname #1@idxfile\endcsname
   \expandafter\immediate\openout \csname #1@idxfile\endcsname #1_html.idx\relax
    \typeout{Writing index file #1_html.idx }%
  \fi}
\catcode`\_=8%
```

\printindex

Patched to use _html filename and \BaseJobname. This will later be patched by the lwarp core.

```
\catcode`\_=12%
\renewcommand{\printindex}[1][\BaseJobname]{\@input@{#1_html.ind}}
\catcode`\_=8%
```

```
\DeclareDocumentCommand{\newblock}{}{}
%
\renewcommand*{\showindexmarks}{}
\renewcommand*{\hideindexmarks}{}

\renewcommand*{\xindyindex}{}
```

§ 695.18 Miscellaneous

```
\renewcommand*{\changemarks}{}
\renewcommand*{\nochangemarks}{}
\renewcommand*{\added}[1]{}
\renewcommand*{\deleted}[1]{}
\renewcommand*{\changed}[1]{}

\renewcommand*{\showtrimsoff}{}
\renewcommand*{\showtrimson}{}
\renewcommand*{\trimXmarks}{}
\renewcommand*{\trimLmarks}{}
```

```
\renewcommand*{\trimFrame}{}
\renewcommand*{\trimNone}{}
\renewcommand*\trimmarkscolor{}
\renewcommand*{\trimmarks}{}
\renewcommand*{\tmarktl}{}
\renewcommand*{\tmarktr}{}
\renewcommand*{\tmarkbr}{}
\renewcommand*{\tmarkbl}{}
\renewcommand*{\tmarktm}{}
\renewcommand*{\tmarkmr}{}
\renewcommand*{\tmarkbm}{}
\renewcommand*{\tmarkml}{}
\renewcommand*{\trimmark}{}
\renewcommand*{\quarkmarks}{}
\renewcommand*{\registrationColour}[1]{}

\renewcommand*{\leavespergathering}[1]{}

\renewcommand*{\noprelistbreak}{}

\renewcommand*{\cleartorecto}{}
\renewcommand*{\cleartoverso}{}

\renewenvironment{vplace}[1][]{}{}
```

§ 695.19 ccaption emulation

```
\renewcommand*{\captiondelim}[1]{\renewcommand*{\CaptionSeparator}{#1}}
\renewcommand*{\captionnamefont}[1]{}
\renewcommand*{\captiontitlefont}[1]{}
\renewcommand*{\flushleftright}{}
\renewcommand*{\centerlastline}{}
\renewcommand*{\captionstyle}[2][]{}
\DeclareDocumentCommand{\captionwidth}{m}{}
\renewcommand*{\changecaptionwidth}{}
\renewcommand*{\normalcaptionwidth}{}
\renewcommand*{\hangcaption}{}
\renewcommand*{\indentcaption}[1]{}
\renewcommand*{\normalcaption}{}
\renewcommand{\precaption}[1]{}
\renewcommand{\postcaption}[1]{}
\renewcommand{\midbicaption}[1]{}
\renewcommand{\contcaption}[1]{%
%   \ContinuedFloat%
%   \caption{#1}%
    \begin{LWR@figcaption}% later becomes \caption*
    \LWR@isolate{\@nameuse{\@captype name}}~%
    \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#1}%
    \end{LWR@figcaption}%
}

\newlength{\abovelegendskip}
\setlength{\abovelegendskip}{0.5\baselineskip}
\newlength{\belowlegendskip}
\setlength{\belowlegendskip}{\abovelegendskip}
```

The extra \\ here forces a <br> in HTML when \legend is used in a \marginpar.

```
\renewcommand{\legend}[1]{\begin{center}#1\\\end{center}}

```

```
\renewcommand{\namedlegend}[2][]{%
    \begin{center}
    \@nameuse{fleg\@captype}\CaptionSeparator#2\\
    \end{center}
    \@nameuse{flegtoc\@captype}{#1}
}
```

`\flegtable`, `\flegfigure`, `\flegtoctable`, `\flegtocfigure` are defined by memoir using `\newfloat`. These are defined with an `@` in ccaption.

```
\renewcommand{\newfixedcaption}[3][\caption]{%
  \renewcommand{#2}{\def\@captype{#3}#1}}
\renewcommand{\renewfixedcaption}[3][\caption]{%
  \renewcommand{#2}{\def\@captype{#3}#1}}
\renewcommand{\providefixedcaption}[3][\caption]{%
  \providecommand{#2}{\def\@captype{#3}#1}}

\renewcommand{\bitwonumcaption}[6][]{%
    \ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
    \addtocounter{\@captype}{-1}%
    \begingroup%
    \csdef{\@captype name}{#4}%
    \ifblank{#5}{\caption{#6}}{\caption[#5]{#6}}%
    \endgroup%
    \ifblank{#1}{}{\label{#1}}%
}

\LetLtxMacro\bionenumcaption\bitwonumcaption% todo

\renewcommand{\bicaption}[5][]{%
    \ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
    \begin{LWR@figcaption}% later becomes \caption*
    \LWR@isolate{#4} % space
    \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#5}%
    \end{LWR@figcaption}%
    \ifblank{#1}{}{\label{#1}}%
}

\renewcommand{\bicontcaption}[3]{%
    \contcaption{#1}%
    \begingroup%
    \csdef{\@captype name}{#2}%
    \contcaption{#3}%
    \endgroup%
}
```

Only in ccaption, not in memoir:

```
% \LetLtxMacro\longbitwonumcaption\bitwonumcaption%
% \LetLtxMacro\longbionenumcaption\bitwonumcaption%
% \LetLtxMacro\longbicaption\bicaption%
```

Patches for subfloats to support additional lwarp labels:

```
\renewcommand{\@memsubbody}{%
  \bgroup
  \let\label=\memsub@label
  \ifdonemaincaption\else
    \advance\csname c@\@captype\endcsname\@ne
  \fi
```

```
  % \refstepcounter{sub\@captype}\@contkeep%
%   \leavevmode%            lwarp
  \@ifnextchar [%
    {\@memsubfig}%
    {\@memsubfig[\@empty]}}

\renewcommand{\@memcontsubbody}{%
  \bgroup
  \let\label=\memsub@label
  \@contset
  % \refstepcounter{sub\@captype}\@contkeep%
%   \leavevmode%            lwarp
  \@ifnextchar [%
    {\@memsubfig}%
    {\@memsubfig[\@empty]}}


\long\def\@memsubfloat#1[#2][#3]#4{%
%   \@tempcnta=\@ne
%   \if@tightsubcap
%     \if@minipage
%       \@tempcnta=\z@
%     \else
%       \ifdim\lastskip=\z@
%         \@tempcnta=\@ne
%       \else
%         \@tempcnta=\tw@
%       \fi
%     \fi
%   \fi
%   \if@contbotsub
%     \def\subfig@top{\subfloattopskip}%
%     \def\subfig@bottom{\subfloatbottomskip}%
%   \else
%     \def\subfig@top{\subfloatbottomskip}%
%     \def\subfig@bottom{\subfloattopskip}%
%   \fi
%   \setbox\@tempboxa \hbox{#4}%
%   \@tempdima=\wd\@tempboxa
%   \vbox
  \bgroup%
    \mem@step@subcounter%
%     \vbox
    \LWR@stoppars%
    \minipagefullwidth%                 lwarp
    \begin{minipage}{\linewidth}%       lwarp
    \bgroup
%     \ifcase\@tempcnta
%       \@minipagefalse
%     \or
%       \vspace{\subfig@top}
%     \or
%       \ifdim \lastskip=\z@ \else
%         \@tempskipb\subfig@top\@xaddvskip
%       \fi
%     \fi
    \if@contbotsub
      #4% \box\@tempboxa
      \egroup
      \ifx \@empty#3\relax \else
```

```
%         \vskip\subfloatcapskip
        \@memsubcaption{#1}{#2}{#3}%
      \fi
    \else
      \ifx \@empty#3\relax \else
        \@memsubcaption{#1}{#2}{#3}%
%         \vskip\subfloatcapskip
%         \vskip\subfloatcaptopadj
      \fi\egroup
      #4% \box\@tempboxa
    \fi
%     \vspace{\subfig@bottom}
    \end{minipage}%                    lwarp
    \LWR@startpars%                    lwarp
  \egroup
\egroup
}
```

§ 695.20 Final patchwork

```
\newlistof{tableofcontents}{toc}{\contentsname}
\newlistof{listoffigures}{lof}{\listfigurename}
\newlistof{listoftables}{lot}{\listtablename}
```

File 587 **lwarp-common-multimedia.sty**

§ 696 Package common-multimedia

lwarp-common-multimedia (*Pkg*) Common code for multimedia, movie15, and media9.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

for HTML output:

```
\ProvidesPackage{lwarp-common-multimedia}[2019/04/22]

\RequirePackage{xkeyval}

\define@key{LWR@multimedia}{width}{\setlength{\LWR@multimedia@width}{#1}}
\define@key{LWR@multimedia}{height}{\setlength{\LWR@multimedia@height}{#1}}
\define@key{LWR@multimedia}{totalheight}{\setlength{\LWR@multimedia@height}{#1}}

\newlength{\LWR@multimedia@width}
\newlength{\LWR@multimedia@height}
\newlength{\LWR@multimedia@maxdimension}
```

\LWR@multimedia@printsize

Proportional to \linewidth and the viewport's smaller dimension. This scales each object such that it will always fit on the screen, even if a tall or wide object inside a tall or wide viewport.

```
\newcommand*{\LWR@multimedia@printsize}{%
    \setlength{\LWR@multimedia@maxdimension}{%
        \maxof%
            {\linewidth}%
            {\maxof{\LWR@multimedia@width}{\LWR@multimedia@height}}%
    }%
   \setlength{\LWR@multimedia@maxdimension}{1.1\LWR@multimedia@maxdimension}%
    \ifdimgreater{\LWR@multimedia@width}{0pt}{%
        width:%
            \LWR@printpercentlength%
                {\LWR@multimedia@width}%
                {\LWR@multimedia@maxdimension}vmin ; % space
    }{}%
    \ifdimgreater{\LWR@multimedia@height}{0pt}{%
        height:%
            \LWR@printpercentlength%
                {\LWR@multimedia@height}%
                {\LWR@multimedia@maxdimension}vmin ; % space
    }{}%
}
```

\LWR@multimedia@fileAV {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates a video or audio from a file. The 2019/10 update of the LaTeX kernel may cause extra quotes to be added in the filenames. They are removed here.

```
\newcommand*{\LWR@multimedia@fileAV}[4]{%
\IfFileExists{#2}{% also sets \@filef@und
\StrSubstitute[100]{\@filef@und}{"}{}[\LWR@parsedfilename]%
```

The container is sized as desired.

```
    \ifstrequal{#3}{audio}{%
```

```
        \begin{BlockClass}{AVviewport}
    }{%
      \begin{BlockClass}[\LWR@multimedia@printsize\ margin:auto]{AVviewport}
    }
```

Paragraph tags are unnecessary for the A / V tags.

```
    \LWR@stoppars
```

The A / V element is 100% of the container.

```
    \LWR@htmltag{%
        #3\ % space
        \ifstrequal{#3}{audio}{}{%
            width=\textquotedbl{}100\%\textquotedbl\ % space
            height=\textquotedbl{}100\%\textquotedbl\ % space
        }%
        controls%
    }\LWR@orignewline
```

The file source and type:

```
    \LWR@htmltag{%
        source % space
        src=\textquotedbl%
        \LWR@parsedfilename\unskip\textquotedbl\ % space
        type=\textquotedbl{}#4\textquotedbl}
```

The poster text inside paragraph tags, along with a reference to the file.

```
    \LWR@startpars
    \LWR@href{\LWR@parsedfilename}{#1}
    \LWR@stoppars
```

Finish.

```
    \LWR@htmltag{/#3}\LWR@orignewline
    \end{BlockClass}
}{%
    \PackageError{lwarp-common-multimedia}
        {File '#2' not found}
        {Perhaps an incorrect path?}
}%
}
```

`\LWR@multimedia@httpAV` {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates a video or audio from a URL link.

```
\newcommand*{\LWR@multimedia@httpAV}[4]{%
```

The container is sized as desired.

```
    \ifstrequal{#3}{audio}{%
        \begin{BlockClass}{AVviewport}
    }{%
      \begin{BlockClass}[\LWR@multimedia@printsize\ margin:auto]{AVviewport}
    }
```

Paragraph tags are unnecessary for the A / V tags.

```
    \LWR@stoppars
```

The A / V element is 100% of the container.

```
    \LWR@htmltag{%
        #3\ % space
```

```
        \ifstrequal{#3}{audio}{}{%
            width=\textquotedbl{}100\%\textquotedbl\ % space
            height=\textquotedbl{}100\%\textquotedbl\ controls%
        }%
    }\LWR@orignewline
```

The file source and type:

```
    \LWR@htmltag{%
        source % space
        src=\textquotedbl#2\textquotedbl\ % space
        type=\textquotedbl#4\textquotedbl}
```

The poster text inside paragraph tags, along with a reference to the URL.

```
    \LWR@startpars
    \LWR@href{#2}{#1}
    \LWR@stoppars
```

Finish.

```
    \LWR@htmltag{/#3}\LWR@orignewline
    \end{BlockClass}
}
```

`\LWR@multimedia@AV` {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates an audio or video from a file or a URL.

```
\newcommand*{\LWR@multimedia@AV}[4]{%
    \IfBeginWith{#2}{http}%
        {\LWR@multimedia@httpAV{#1}{#2}{#3}{#4}}%
        {%
            \IfBeginWith{#2}{HTTP}%
                {\LWR@multimedia@httpAV{#1}{#2}{#3}{#4}}%
                {\LWR@multimedia@fileAV{#1}{#2}{#3}{#4}}%
        }%
}
```

`\LWR@multimedia@embed` {⟨*poster text*⟩} {⟨*URL or filename*⟩} {⟨*mime type*⟩}

Embeds multimedia of an arbitrary type. The poster text is not used, as it would appear along with the video if the <embed> element is supported.

```
\newcommand*{\LWR@multimedia@embed}[3]{%
    \begin{BlockClass}[width:100\%]{AVviewport}%
    \LWR@stoppars
        \LWR@htmltag{%
            embed % space
            \ifblank{#3}{}{type=\textquotedbl#3\textquotedbl\ }%
        style=\textquotedbl\LWR@multimedia@printsize\ margin:auto\textquotedbl\ % space
            src=\textquotedbl#2\textquotedbl\ % space
        }%
    \LWR@startpars
    \end{BlockClass}
}
```

`\LWR@multimedia@percenterror` Error message if the comment character is used among the arguments of \LWR@multimediab.

```
\newcommand*{\LWR@multimedia@percenterror}{%
    \PackageError{lwarp-media9}
    {%
```

```
        Do not use a percent comment between\MessageBreak
        \protect\includemedia\space arguments%
    }
    {%
        Percent is changed to a regular character\MessageBreak
        to allow its use inside a URL.%
    }
}
```

\LWR@multimediab [⟨*options*⟩] {⟨*poster text*⟩} {⟨*filename*⟩}

Creates multimedia. Examines the file extension to determine the type. If not a supported type, creates an embedded object if it has a URL. If neither, create a link to the unsupported object.

```
\newcommand*{\LWR@multimediab}[3][]{%
```

Error if the percent character appears among the arguments. This could happen since the comment character has been temporarily disabled, for use in a URL.

```
    \if#1\@percentchar\LWR@multimedia@percenterror\fi%
    \if#2\@percentchar\LWR@multimedia@percenterror\fi%
    \if#3\@percentchar\LWR@multimedia@percenterror\fi%
```

Paragraph handling:

```
    \LWR@stoppars%
```

Record the desired size.

```
    \setlength{\LWR@multimedia@width}{0pt}%
    \setlength{\LWR@multimedia@height}{0pt}%
    \setkeys*{LWR@multimedia}{#1}%
```

If a known A/V type, create an HTML5 <video> or <audio>.

```
    \IfEndWith{#3}{.mp4}{\LWR@multimedia@AV{#2}{#3}{video}{video/mp4}}{%
    \IfEndWith{#3}{.MP4}{\LWR@multimedia@AV{#2}{#3}{video}{video/mp4}}{%
    \IfEndWith{#3}{.mp3}{\LWR@multimedia@AV{#2}{#3}{audio}{audio/mpeg}}{%
    \IfEndWith{#3}{.MP3}{\LWR@multimedia@AV{#2}{#3}{audio}{audio/mpeg}}{%
```

If an arbitrary URL, embed it.

```
    \IfBeginWith{#3}{http}{\LWR@multimedia@embed{#2}{#3}{}}{%
    \IfBeginWith{#3}{HTTP}{\LWR@multimedia@embed{#2}{#3}{}}{%
    \IfBeginWith{#3}{ftp}{\LWR@multimedia@embed{#2}{#3}{}}{%
    \IfBeginWith{#3}{FTP}{\LWR@multimedia@embed{#2}{#3}{}}{%
```

If unknown, create a link to it.

```
        \LWR@href{#3}{#2}% unknown format
    }}}}}}}}%
```

Paragraph handling:

```
    \LWR@startpars%
    \endgroup%
}
```

Catcodes which may apper in a URL.

```
\newrobustcmd*{\LWR@multimedia}{%
    \begingroup%
    \LWR@linkmediacatcodes%
    \LWR@multimediab%
}
```

File 588 **lwarp-common-mathjax-letters.sty**

§ 697 Package

common-mathjax-letters

lwarp-common-mathjax-letters (*Pkg*)

Common code used by a number of packages to generate Greek math characters for MathJax.

for HTML output:

```
\ProvidesPackage{lwarp-common-mathjax-letters}[2020/08/10]
```

\LWR@mathjax@addletter

* {⟨*2: capitalize name?*⟩} {⟨*3: prefix*⟩} {⟨*4: postfix*⟩} {⟨*5: name*⟩} {⟨*6: unicode*⟩}

Star to italicize the result, used when the unicode character does not exist.

```
\begin{warpMathJax}

\NewDocumentCommand{\LWR@mathjax@addletter}{s m m m m m}{
    \IfBooleanTF{#2}%
        {\edef\LWR@tempone{\LWRtexttitlecase{#5}}}%
        {\edef\LWR@tempone{#5}}%
    \xdef\LWR@customizedMathJax{%
        \LWR@customizedMathJax%
        \LWRbackslash(%
        \LWRbackslash def\LWRbackslash%
        #3% prefix
        \LWR@tempone%name
        #4% postfix
        \LWRleftbrace%
    }%
    \IfBooleanTF{#1}{%
        \xdef\LWR@customizedMathJax{%
            \LWR@customizedMathJax%
            \LWRbackslash mathit\LWRleftbrace%
            \LWRbackslash unicode\LWRleftbrace x#6\LWRrightbrace%
            \LWRrightbrace%
        }%
    }{%
        \xdef\LWR@customizedMathJax{%
            \LWR@customizedMathJax%
            \LWRbackslash unicode\LWRleftbrace x#6\LWRrightbrace%
        }%
    }%
    \xdef\LWR@customizedMathJax{%
        \LWR@customizedMathJax%
        \LWRrightbrace\LWRbackslash)\par%
    }%
}
```

\LWR@mathjax@addgreek@l@up

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

Star to capitalize the macro names.

Adds `\CustomizeMathjax` expressions to define a set of macros for Greek letters, lowercase upright.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@l@up}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{03B1}
```

```
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{03B2}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varbeta}{03D0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{03B3}
    \LWR@mathjax@addletter{#1}{#2}{#3}{digamma}{03DD}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{03B4}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{03F5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varepsilon}{03B5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{03B6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{03B7}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{03B8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{03D1}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{03B9}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{03BA}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varkappa}{03F0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{03BB}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{03BC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{03BD}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{03BE}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{03BF}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{03C0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{03D6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{03C1}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varrho}{03F1}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{03C3}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varsigma}{03C2}
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{03C4}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{03C5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{03D5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varphi}{03C6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{03C7}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{03C8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{03C9}
}
```

\LWR@mathjax@addgreek@u@up

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

Star to capitalize the macro names.

Adds `\CustomizeMathjax` expressions to define a set of macros for Greek letters, uppercase upright.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@u@up}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{0391}
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{0392}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{0393}
    \LWR@mathjax@addletter{#1}{#2}{#3}{digamma}{03DC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{0394}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{0395}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{0396}
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{0397}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{0398}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{03F4}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{0399}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{039A}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{039B}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{039C}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{039D}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{039E}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{039F}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{03A0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{03D6}
```

```
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{03A1}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{03A3}
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{03A4}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{03A5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{03A6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{03A7}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{03A8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{03A9}
}
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@l@it

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, lowercase italic.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@l@it}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D6FC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D6FD}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varbeta}{03D0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D6FE}
    \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DD}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D6FF}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D716}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varepsilon}{1D700}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D701}
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D702}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D703}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D717}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D704}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D705}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varkappa}{1D718}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D706}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D707}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D708}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D709}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D70A}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D70B}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{1D71B}
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D70C}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varrho}{1D71A}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D70E}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varsigma}{1D70D}
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D70F}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D710}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D719}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varphi}{1D711}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D712}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D713}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D714}
}
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@u@it

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, uppercase italic.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@u@it}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D6E2}
```

```
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D6E3}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D6E4}
    \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D6E5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D6E6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D6E7}
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D6E8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D6E9}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D6F3}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D6EA}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D6EB}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D6EC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D6ED}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D6EE}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D6EF}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D6F0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D6F1}
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D6F2}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D6F4}
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D6F5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D6F6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D6F7}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D6F8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D6F9}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D6FA}
}
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@l@bfit

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, lowercase boldface italic.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@l@bfit}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D736}
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D737}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varbeta}{03D0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D738}
    \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DD}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D739}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D750}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varepsilon}{1D73A}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D73B}
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D73C}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D73D}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D751}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D73E}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D73F}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varkappa}{1D752}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D740}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D741}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D742}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D743}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D744}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D745}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{1D755}
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D746}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varrho}{1D754}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D748}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varsigma}{1D747}
```

```
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D749}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D74A}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D753}
    \LWR@mathjax@addletter{#1}{#2}{#3}{varphi}{1D74B}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D74C}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D74D}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D74E}
}
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@u@bfit

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, uppercase boldface italic.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@u@bfit}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D71C}
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D71D}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D71E}
    \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D71F}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D720}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D721}
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D722}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D723}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D72D}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D724}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D725}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D726}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D727}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D728}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D729}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D72A}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D72B}
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D72C}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D72E}
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D72F}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D730}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D731}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D732}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D733}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D734}
}
```

\LWR@mathjax@addgreek@u@bfup is not needed.

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@u@bfup

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, uppercase boldface upright.

```
\NewDocumentCommand{\LWR@mathjax@addgreek@u@bfup}{s m m}{
    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D6A8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D6A9}
    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D6AA}
    \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D6AB}
    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D6AC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D6AD}
```

```
    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D6AE}
    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D6AF}
    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D6B9}
    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D6B0}
    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D6B1}
    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D6B2}
    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D6B3}
    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D6B4}
    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D6B5}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D6B6}
    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D6B7}
    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D6B8}
    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D6BA}
    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D6BB}
    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D6BC}
    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D6BD}
    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D6BE}
    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D6BF}
    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D6C0}
}
```

{⟨*prefix*⟩}

`\LWR@mathjax@addlatin@u@bfit` Adds `\CustomizeMathjax` expressions to define a set of macros for bold-face italic Latin letters, uppercase and lowercase.

```
\NewDocumentCommand{\LWR@mathjax@addlatin@u@bfit}{m}{
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{A}{1D468}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{B}{1D469}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{C}{1D46A}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{D}{1D46B}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{E}{1D46C}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{F}{1D46D}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{G}{1D46E}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{H}{1D46F}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{I}{1D470}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{J}{1D471}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{K}{1D472}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{L}{1D473}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{M}{1D474}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{N}{1D475}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{O}{1D476}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{P}{1D477}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{Q}{1D478}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{R}{1D479}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{S}{1D47A}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{T}{1D47B}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{U}{1D47C}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{V}{1D47D}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{W}{1D47E}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{X}{1D47F}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{Y}{1D480}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{Z}{1D481}
}
```

{⟨*prefix*⟩}

`\LWR@mathjax@addlatin@l@bfit` Adds `\CustomizeMathjax` expressions to define a set of macros for bold-face italic Latin letters, uppercase and lowercase.

```
\NewDocumentCommand{\LWR@mathjax@addlatin@l@bfit}{m}{
```

```
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{a}{1D482}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{b}{1D483}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{c}{1D484}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{d}{1D485}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{e}{1D486}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{f}{1D487}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{g}{1D488}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{h}{1D489}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{i}{1D48A}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{j}{1D48B}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{k}{1D48C}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{l}{1D48D}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{m}{1D48E}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{n}{1D48F}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{o}{1D490}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{p}{1D491}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{q}{1D492}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{r}{1D493}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{s}{1D494}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{t}{1D495}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{u}{1D496}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{v}{1D497}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{w}{1D498}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{x}{1D499}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{y}{1D49A}
    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{z}{1D49B}
}

\end{warpMathJax}
```

File 589 **lwarp-common-mathjax-newpxtxmath.sty**

§ 698 Package

common-mathjax-newpxtxmath

(*Emulates or patches code by* Michael Sharpe.)

lwarp-common-mathjax-newpxtxmath (*Pkg*)

Common code used by newpxmath, newtxmath, and newtxsf for MathJax.

for HTML output:

```
\ProvidesPackage{lwarp-common-mathjax-newpxtxmath}[2020/09/20]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-nonunicode}
\LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\fAlt}{f}}
\CustomizeMathJax{\newcommand{\rhoAlt}{\rho}}

\CustomizeMathJax{\newcommand{\imathscr}{\mathord{\mathscr{i}}}}
\CustomizeMathJax{\newcommand{\jmathscr}{\mathord{\mathscr{j}}}}
```

`lwarp_mathjax.txt` adds `\left`/`\right` support for delimiters.

```
\CustomizeMathJax{\let\llbracket\lBrack}
\CustomizeMathJax{\let\rrbracket\rBrack}
```

```
13
14 \CustomizeMathJax{\let\smlbrace\{}
15 \CustomizeMathJax{\let\smrbrace\}}
16 \CustomizeMathJax{\newcommand{\Perp}{\mathrel{\unicode{x02AEB}}}}
17 \CustomizeMathJax{\newcommand{\nPerp}{\mathrel{\not{\!\unicode{x02AEB}}}}}
18 \CustomizeMathJax{\newcommand{\Zbar}{\mathord{\unicode{x01B5}}}}
19 \CustomizeMathJax{\newcommand{\Angstrom}{\mathord{\unicode{x212B}}}}
20 \CustomizeMathJax{\newcommand{\Euler}{\mathord{\unicode{x2107}}}}
21 \CustomizeMathJax{\newcommand{\transp}{\mathord{\unicode{xFF34}}}}
22 \CustomizeMathJax{\newcommand{\hermtransp}{\mathord{\unicode{xFF28}}}}
23 \CustomizeMathJax{\let\htransp=\hermtransp}
24 \CustomizeMathJax{\newcommand{\circledplus}{\mathbin{\unicode{x2295}}}}
25 \CustomizeMathJax{\newcommand{\circledminus}{\mathbin{\unicode{x2296}}}}
26 \CustomizeMathJax{\newcommand{\circledtimes}{\mathbin{\unicode{x2297}}}}


27 \CustomizeMathJax{\newcommand{\circledslash}{\mathbin{\unicode{x2298}}}}
28 %
29 \CustomizeMathJax{\newcommand{\circleddot}{\mathbin{\unicode{x2299}}}}
30 \CustomizeMathJax{\let\overgroup\overparen}
31 \CustomizeMathJax{\let\overgroupra\overrightarrow}
32 \CustomizeMathJax{\let\undergroup\underparen}
33 \CustomizeMathJax{\let\undergroupla\underleftarrow}
34 \CustomizeMathJax{\newcommand{\widering}[1]{%
35     \stackrel{\unicode{x2218}}{\overgroup{#1}}%
36 }}
37 \CustomizeMathJax{\let\widearc\overparen}
38 \CustomizeMathJax{\let\wideOarc\overrightarrow}
39 \CustomizeMathJax{\newcommand{\LWRvvstar}[2]{{\overrightarrow{#1}_{#2}}}}
40 \CustomizeMathJax{\newcommand{\vv}{\ifstar\LWRvvstar\overrightarrow}}
41 %
42 \CustomizeMathJax{\let\smallintsl\smallint}
43 \CustomizeMathJax{\newcommand{\smalliintsl}{\mathop{\unicode{x222C}}\limits}}
44 \CustomizeMathJax{\newcommand{\smalliiintsl}{\mathop{\unicode{x222D}}\limits}}
45 \CustomizeMathJax{\newcommand{\smalliiiintsl}{\mathop{\unicode{x2A0C}}\limits}}
46 \CustomizeMathJax{\newcommand{\smallointsl}{\mathop{\unicode{x222E}}\limits}}
47 \CustomizeMathJax{\newcommand{\smalloiintsl}{\mathop{\unicode{x222F}}\limits}}
48 \CustomizeMathJax{\newcommand{\smalloiiintsl}{\mathop{\unicode{x2230}}\limits}}
49 \CustomizeMathJax{\newcommand{\smallvarointclockwisesl}{%
50     \mathop{\unicode{x2232}}\limits%
51 }}
52 \CustomizeMathJax{\newcommand{\smallointctrclockwisesl}{%
53     \mathop{\unicode{x2233}}\limits%
54 }}
55 \CustomizeMathJax{\newcommand{\smallsumintsl}{\mathop{\unicode{x2A0B}}\limits}}
56 \CustomizeMathJax{\newcommand{\smallfintsl}{\mathop{\unicode{x2A0F}}\limits}}
57 \CustomizeMathJax{\newcommand{\smallsqintsl}{\mathop{\unicode{x2A16}}\limits}}
58 %
59 \CustomizeMathJax{\let\smallintup\smallint}
60 \CustomizeMathJax{\newcommand{\smalliintup}{\mathop{\unicode{x222C}}\limits}}
61 \CustomizeMathJax{\newcommand{\smalliiintup}{\mathop{\unicode{x222D}}\limits}}
62 \CustomizeMathJax{\newcommand{\smalliiiintup}{\mathop{\unicode{x2A0C}}\limits}}
63 \CustomizeMathJax{\newcommand{\smallointup}{\mathop{\unicode{x222E}}\limits}}
64 \CustomizeMathJax{\newcommand{\smalloiintup}{\mathop{\unicode{x222F}}\limits}}
65 \CustomizeMathJax{\newcommand{\smalloiiintup}{\mathop{\unicode{x2230}}\limits}}
66 \CustomizeMathJax{\newcommand{\smallvarointclockwiseup}{%
67     \mathop{\unicode{x2232}}\limits%
68 }}
69 \CustomizeMathJax{\newcommand{\smallointctrclockwiseup}{%
70     \mathop{\unicode{x2233}}\limits%
71 }}
```

```
\CustomizeMathJax{\newcommand{\smallsumintup}{\mathop{\unicode{x2A0B}}\limits}}
\CustomizeMathJax{\newcommand{\smallfintup}{\mathop{\unicode{x2A0F}}\limits}}
\CustomizeMathJax{\newcommand{\smallsqintup}{\mathop{\unicode{x2A16}}\limits}}
%
\CustomizeMathJax{\newcommand{\iint}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiint}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\iiiint}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
\CustomizeMathJax{\newcommand{\varointclockwise}{\mathop{\unicode{x2232}}\limits}}
\CustomizeMathJax{\newcommand{\ointctrclockwise}{\mathop{\unicode{x2233}}\limits}}
\CustomizeMathJax{\newcommand{\sumint}{\mathop{\unicode{x2A0B}}\limits}}
\CustomizeMathJax{\newcommand{\fint}{\mathop{\unicode{x2A0F}}\limits}}
\CustomizeMathJax{\newcommand{\sqint}{\mathop{\unicode{x2A16}}\limits}}
%
\CustomizeMathJax{\let\intsl\int}
\CustomizeMathJax{\newcommand{\iintsl}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiintsl}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\iiiintsl}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\let\ointsl\oint}
\CustomizeMathJax{\newcommand{\oiintsl}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiintsl}{\mathop{\unicode{x2230}}\limits}}
\CustomizeMathJax{\newcommand{\varointclockwisesl}{\mathop{\unicode{x2232}}\limits}}
\CustomizeMathJax{\newcommand{\ointctrclockwisesl}{\mathop{\unicode{x2233}}\limits}}
\CustomizeMathJax{\newcommand{\sumintsl}{\mathop{\unicode{x2A0B}}\limits}}
\CustomizeMathJax{\newcommand{\fintsl}{\mathop{\unicode{x2A0F}}\limits}}
\CustomizeMathJax{\newcommand{\sqintsl}{\mathop{\unicode{x2A16}}\limits}}
%
\CustomizeMathJax{\let\intup\int}
\CustomizeMathJax{\newcommand{\iintup}{\mathop{\unicode{x222C}}\limits}}
\CustomizeMathJax{\newcommand{\iiintup}{\mathop{\unicode{x222D}}\limits}}
\CustomizeMathJax{\newcommand{\iiiintup}{\mathop{\unicode{x2A0C}}\limits}}
\CustomizeMathJax{\let\ointup\oint}
\CustomizeMathJax{\newcommand{\oiintup}{\mathop{\unicode{x222F}}\limits}}
\CustomizeMathJax{\newcommand{\oiiintup}{\mathop{\unicode{x2230}}\limits}}
\CustomizeMathJax{\newcommand{\varointclockwiseup}{%
    \mathop{\unicode{x2232}}\limits%
}}
\CustomizeMathJax{\newcommand{\ointctrclockwiseup}{%
    \mathop{\unicode{x2233}}\limits%
}}
\CustomizeMathJax{\newcommand{\sumintup}{\mathop{\unicode{x2A0B}}\limits}}
\CustomizeMathJax{\newcommand{\fintup}{\mathop{\unicode{x2A0F}}\limits}}
\CustomizeMathJax{\newcommand{\sqintup}{\mathop{\unicode{x2A16}}\limits}}
%
\CustomizeMathJax{\newcommand{\bigcupdot}{\mathop{\unicode{x2A03}}}}
\CustomizeMathJax{\newcommand{\bigcupplus}{\mathop{\unicode{x2A04}}}}
\CustomizeMathJax{\newcommand{\bigsqcap}{\mathop{\unicode{x2A05}}}}
%


%
\CustomizeMathJax{\newcommand{\bigtimes}{\mathop{\unicode{x2A09}}}}
\CustomizeMathJax{\let\varprod\bigtimes}
%


\CustomizeMathJax{\newcommand{\mappedfrom}{\mathrel{\unicode{x021A4}}}}
\CustomizeMathJax{\let\mappedfromchar\mappedfrom}
\CustomizeMathJax{\newcommand{\mapsfrom}{\mathrel{\unicode{x021A4}}}}
\CustomizeMathJax{\newcommand{\longmappedfrom}{\mathrel{\unicode{x027FB}}}}
```

```
%
\CustomizeMathJax{\newcommand{\Mapsto}{\mathrel{\unicode{x02907}}}}
\CustomizeMathJax{\let\Mapstochar\Mapsto}
\CustomizeMathJax{\newcommand{\Longmapsto}{\mathrel{\unicode{x027FE}}}}
\CustomizeMathJax{\newcommand{\Mappedfrom}{\mathrel{\unicode{x02906}}}}
\CustomizeMathJax{\let\Mappedfromchar\Mappedfrom}
\CustomizeMathJax{\newcommand{\Mapsfrom}{\mathrel{\unicode{x02906}}}}
\CustomizeMathJax{\newcommand{\Longmappedfrom}{\mathrel{\unicode{x27FD}}}}
%


\CustomizeMathJax{\newcommand{\medcirc}{\mathbin{\unicode{x025CB}}}}
\CustomizeMathJax{\newcommand{\medbullet}{\mathbin{\unicode{x025CF}}}}
\CustomizeMathJax{\newcommand{\varparallel}{\mathrel{\unicode{x02AFD}}}}
\CustomizeMathJax{\newcommand{\varparallelinv}{\mathrel{\unicode{x244A}}}}
\CustomizeMathJax{\newcommand{\nvarparallel}{%
    \mathrel{\LWRoverlaysymbols{-}{\unicode{x02AFD}}}%
}}
\CustomizeMathJax{\newcommand{\nvarparallelinv}{%
    \mathrel{\LWRoverlaysymbols{-}{\unicode{x244A}}}%
}}
%


\CustomizeMathJax{\newcommand{\coloneq}{\mathrel{\unicode{x02254}}}}
\CustomizeMathJax{\newcommand{\eqcolon}{\mathrel{\unicode{x02255}}}}
%
\CustomizeMathJax{\newcommand{\VDash}{\mathrel{\unicode{x22AB}}}}


%
\CustomizeMathJax{\newcommand{\preceqq}{\mathrel{\unicode{x02AB3}}}}
\CustomizeMathJax{\newcommand{\succeqq}{\mathrel{\unicode{x02AB4}}}}
%

\CustomizeMathJax{\newcommand{\nprecsim}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227E}}}%
}}
\CustomizeMathJax{\newcommand{\nsuccsim}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227F}}}%
}}
\CustomizeMathJax{\newcommand{\nlesssim}{\mathrel{\unicode{x02274}}}}
\CustomizeMathJax{\newcommand{\ngtrsim}{\mathrel{\unicode{x02275}}}}
%


\CustomizeMathJax{\newcommand{\nsubset}{\mathrel{\unicode{x02284}}}}
\CustomizeMathJax{\newcommand{\nsupset}{\mathrel{\unicode{x02285}}}}
\CustomizeMathJax{\newcommand{\notni}{\mathrel{\unicode{x220C}}}}
\CustomizeMathJax{\let\notowns\notni}
%
\CustomizeMathJax{\newcommand{\nlessapprox}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02A85}}}%
}}
\CustomizeMathJax{\newcommand{\ngtrapprox}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02A86}}}%
}}
%
\CustomizeMathJax{\newcommand{\npreccurlyeq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227C}}}%
}}
```

```
\CustomizeMathJax{\newcommand{\nsucccurlyeq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227D}}}%
}}
\CustomizeMathJax{\newcommand{\ngtrless}{\mathrel{\unicode{x02279}}}}
\CustomizeMathJax{\newcommand{\nlessgtr}{\mathrel{\unicode{x2278}}}}
\CustomizeMathJax{\newcommand{\nbumpeq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0224F}}}%
}}
\CustomizeMathJax{\newcommand{\nBumpeq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0224E}}}%
}}
%
\CustomizeMathJax{\newcommand{\nbacksim}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0223D}}}%
}}
\CustomizeMathJax{\newcommand{\nbacksimeq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x022CD}}}%
}}
\CustomizeMathJax{\newcommand{\nasymp}{\mathrel{\unicode{x226D}}}}
\CustomizeMathJax{\newcommand{\nequiv}{\mathrel{\unicode{x2262}}}}
\CustomizeMathJax{\newcommand{\napprox}{\mathrel{\unicode{x2249}}}}
%
\CustomizeMathJax{\newcommand{\nll}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0226A}}}%
}}
\CustomizeMathJax{\newcommand{\ngg}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0226B}}}%
}}
\CustomizeMathJax{\newcommand{\nthickapprox}{%
    \mathrel{\LWRoverlaysymbols{/}{{\mathbf{\unicode{x02248}}}}}%
}}
\CustomizeMathJax{\newcommand{\napproxeq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0224A}}}%
}}
\CustomizeMathJax{\newcommand{\nprecapprox}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB7}}}%
}}
\CustomizeMathJax{\newcommand{\nsuccapprox}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB8}}}%
}}
\CustomizeMathJax{\newcommand{\npreceqq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB3}}}%
}}
\CustomizeMathJax{\newcommand{\nsucceqq}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB4}}}%
}}
\CustomizeMathJax{\newcommand{\nsimeq}{\mathrel{\unicode{x02244}}}}
%
\CustomizeMathJax{\newcommand{\nSubset}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x022D0}}}%
}}
\CustomizeMathJax{\newcommand{\nSupset}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x022D1}}}%
}}
\CustomizeMathJax{\newcommand{\nsqsubseteq}{\mathrel{\unicode{x022E2}}}}
\CustomizeMathJax{\newcommand{\nsqsupseteq}{\mathrel{\unicode{x022E3}}}}
%
\CustomizeMathJax{\newcommand{\coloneqq}{\mathrel{\unicode{x02254}}}}
\CustomizeMathJax{\newcommand{\eqqcolon}{\mathrel{\unicode{x02255}}}}
\CustomizeMathJax{\newcommand{\Coloneqq}{\mathrel{\unicode{x02A74}}}}
```

```
\CustomizeMathJax{\newcommand{\Coloneq}{\mathrel{\unicode{x2237}-}}}
\CustomizeMathJax{\newcommand{\Eqcolon}{\mathrel{-\unicode{x2237}}}}
%
\CustomizeMathJax{\newcommand{\lvec}[1]{%
    \mathord{\overset{\unicode{x02190}}{#1}}%
}}
\CustomizeMathJax{\newcommand{\lrvec}[1]{%
    \mathord{\overset{\unicode{x2194}}{#1}}%
}}
\CustomizeMathJax{\newcommand{\harpoonacc}[1]{%
    \mathord{\overset{\unicode{x021C0}}{#1}}%
}}
\CustomizeMathJax{\newcommand{\lharpoonacc}[1]{%
    \mathord{\overset{\unicode{x021BC}}{#1}}%
}}
\CustomizeMathJax{\newcommand{\lrharpoonacc}[1]{%
    \mathord{\overset{\unicode{x0294E}}{#1}}%
}}
\CustomizeMathJax{\newcommand{\barbar}[1]{\mathord{\overset{=}{#1}}}}
\CustomizeMathJax{\newcommand{\bartilde}[1]{\mathord{\overset{\simeq}{#1}}}}
\CustomizeMathJax{\newcommand{\barhat}[1]{\mathord{\hat{\bar{#1}}}}}
\CustomizeMathJax{\newcommand{\tildebar}[1]{\mathord{\overset{\eqsim}{#1}}}}
\CustomizeMathJax{\newcommand{\tildetilde}[1]{\mathord{\overset{\approx}{#1}}}}
\CustomizeMathJax{\newcommand{\tildehat}[1]{\mathord{\hat{\tilde{#1}}}}}
\CustomizeMathJax{\newcommand{\hatbar}[1]{\mathord{\bar{\hat{#1}}}}}
\CustomizeMathJax{\newcommand{\hattilde}[1]{\mathord{\tilde{\hat{#1}}}}}
\CustomizeMathJax{\newcommand{\hathat}[1]{\mathord{\hat{\hat{#1}}}}}

\CustomizeMathJax{\newcommand{\cdotB}{\mathord{\boldsymbol{\cdot}}}}
\CustomizeMathJax{\newcommand{\cdotBB}{\mathord{\unicode{x2022}}}}
\CustomizeMathJax{\newcommand{\circS}{\boldsymbol{\circ}}}
\CustomizeMathJax{\newcommand{\bulletSSS}{\bullet}}
\CustomizeMathJax{\newcommand{\bulletSS}{\mathord{\unicode{x025CF}}}}
\CustomizeMathJax{\newcommand{\bulletS}{\mathord{\unicode{x02B24}}}}
\CustomizeMathJax{\newcommand{\primeS}{\prime}}

\CustomizeMathJax{\newcommand{\invamp}{\mathbin{\unicode{x0214B}}}}
```

`lwarp_mathjax.txt` adds `\left`/`\right` support for delimiters.

```
\CustomizeMathJax{\newcommand{\Lbag}{\mathopen{\large\unicode{x027C5}}}}
\CustomizeMathJax{\newcommand{\Rbag}{\mathclose{\large\unicode{x027C6}}}}
\CustomizeMathJax{\newcommand{\circledless}{\mathrel{\unicode{x029C0}}}}
\CustomizeMathJax{\newcommand{\circledgtr}{\mathrel{\unicode{x029C1}}}}
\CustomizeMathJax{\newcommand{\circledbslash}{\mathbin{\unicode{x029B8}}}}


\CustomizeMathJax{\newcommand{\lJoin}{\mathrel{\unicode{x22C9}}}}
\CustomizeMathJax{\newcommand{\rJoin}{\mathrel{\unicode{x22CA}}}}
\CustomizeMathJax{\newcommand{\lrJoin}{\mathrel{\unicode{x2A1D}}}}

\CustomizeMathJax{\newcommand{\lrtimes}{\mathrel{\unicode{x2A1D}}}}
\CustomizeMathJax{\newcommand{\Diamondblack}{\mathord{\unicode{x025C6}}}}
\CustomizeMathJax{\newcommand{\nplus}{%
    \mathrel{\LWRoverlaysymbols{+}{\unicode{x02229}}}%
}}
\CustomizeMathJax{\newcommand{\nsqsubset}{%
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x0228F}}}%
}}
\CustomizeMathJax{\newcommand{\nsqsupset}{%
```

```
    \mathrel{\LWRoverlaysymbols{/}{\unicode{x02290}}}%
}}
\CustomizeMathJax{\newcommand{\dasharrow}{\mathrel{\unicode{x021E2}}}}
\CustomizeMathJax{\newcommand{\leftsquigarrow}{\mathrel{\unicode{x021DC}}}}
\CustomizeMathJax{\newcommand{\ntwoheadrightarrow}{\mathrel{\unicode{x02900}}}}
\CustomizeMathJax{\newcommand{\ntwoheadleftarrow}{\mathrel{\unicode{x02B34}}}}
\CustomizeMathJax{\newcommand{\boxast}{\mathbin{\unicode{x029C6}}}}
\CustomizeMathJax{\newcommand{\boxbslash}{\mathbin{\unicode{x29C5}}}}
\CustomizeMathJax{\newcommand{\boxbar}{\mathbin{\unicode{x025EB}}}}
\CustomizeMathJax{\newcommand{\boxslash}{\mathbin{\unicode{x029C4}}}}

\CustomizeMathJax{\newcommand{\varclubsuit}{\mathord{\unicode{x02667}}}}
\CustomizeMathJax{\newcommand{\vardiamondsuit}{\mathord{\unicode{x02666}}}}
\CustomizeMathJax{\newcommand{\varheartsuit}{\mathord{\unicode{x02665}}}}
\CustomizeMathJax{\newcommand{\varspadesuit}{\mathord{\unicode{x02664}}}}

\CustomizeMathJax{\newcommand{\Nearrow}{\mathrel{\unicode{x021D7}}}}
\CustomizeMathJax{\newcommand{\Searrow}{\mathrel{\unicode{x021D8}}}}
\CustomizeMathJax{\newcommand{\Nwarrow}{\mathrel{\unicode{x021D6}}}}
\CustomizeMathJax{\newcommand{\Swarrow}{\mathrel{\unicode{x021D9}}}}
\CustomizeMathJax{\newcommand{\Top}{\mathord{\unicode{x02AEA}}}}
\CustomizeMathJax{\newcommand{\Bot}{\mathord{\unicode{x02AEB}}}}

\CustomizeMathJax{\newcommand{\leadstoext}{\mathrel{\unicode{xFF5E}}}}

\CustomizeMathJax{\newcommand{\sqcupplus}{%
    \mathbin{\LWRoverlaysymbols{+}{\unicode{x02294}}}%
}}
\CustomizeMathJax{\newcommand{\sqcapplus}{%
    \mathbin{\LWRoverlaysymbols{+}{\unicode{x02293}}}%
}}

\CustomizeMathJax{\newcommand{\dlb}{\mathopen{\unicode{x027E6}}}}
\CustomizeMathJax{\newcommand{\drb}{\mathopen{\unicode{x027E7}}}}

\CustomizeMathJax{\newcommand{\varg}{g}}
\CustomizeMathJax{\newcommand{\vary}{y}}
\CustomizeMathJax{\newcommand{\varv}{v}}
\CustomizeMathJax{\newcommand{\varw}{w}}

\CustomizeMathJax{\newcommand{\nexistsAlt}{\mathord{\unicode{x02204}}}}
\CustomizeMathJax{\newcommand{\existsAlt}{\mathord{\unicode{x02203}}}}
\CustomizeMathJax{\newcommand{\forallAlt}{\mathord{\unicode{x02200}}}}
\CustomizeMathJax{\newcommand{\emptysetAlt}{\mathord{\unicode{x02205}}}}

\CustomizeMathJax{\newcommand{\uppartial}{%
    \mathord{\unicode{x02202}}%
}}% not upright

\CustomizeMathJax{\let\varmathbb\mathbb}
\CustomizeMathJax{\let\vmathbb\mathbb}
\CustomizeMathJax{\let\vvmathbb\mathbb}

\CustomizeMathJax{\let\smallprod\prod}
\CustomizeMathJax{\let\smallsum\sum}
\CustomizeMathJax{\let\smallcoprod\coprod}

\CustomizeMathJax{\newcommand{\openbox}{\mathord{\unicode{x25FD}}}}
\CustomizeMathJax{\let\textsquare\openbox}
```

```
\CustomizeMathJax{\let\varemptyset\emptyset}
%
% for newpxmath:
\CustomizeMathJax{\newcommand{\mathsterling}{\mathord{\unicode{x000A3}}}}
\CustomizeMathJax{\newcommand{\mathcent}{\mathord{\unicode{x000A2}}}}

\end{warpMathJax}
```

File 590 **lwarp-common-mathjax-nonunicode.sty**

§ 699 Package common-mathjax-nonunicode

(*Emulates or patches code by* Daniel Flipo, Michael Sharpe.)

lwarp-common-mathjax-nonunicode (*Pkg*) Common code used by newpxmath, newtxmath, newtxsf, kpfonts-otf for MathJax. These are symbols not found in Unicode.

Factored from lwarp-common-mathjax-newpxtxmath.

for HTML output:

```
\ProvidesPackage{lwarp-common-mathjax-nonunicode}[2020/09/20]
```

For MathJax:

```
\LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mmapsto}{\mathrel{\unicode{x021A6}}}}
\CustomizeMathJax{\let\mmapstochar\mmapsto}
\CustomizeMathJax{\newcommand{\longmmapsto}{\mathrel{\unicode{x021A6}}}}
\CustomizeMathJax{\newcommand{\mmappedfrom}{\mathrel{\unicode{x021A4}}}}
\CustomizeMathJax{\let\mmappedfromchar\mmappedfrom}
\CustomizeMathJax{\newcommand{\longmmappedfrom}{\mathrel{\unicode{x021A4}}}}
\CustomizeMathJax{\let\mmapsfrom\mmappedfrom}% from kpfonts-otf
\CustomizeMathJax{\let\longmmapsfrom\longmmappedfrom}% from kpfonts-otf

\CustomizeMathJax{\newcommand{\Mmapsto}{\mathrel{\unicode{x02907}}}}
\CustomizeMathJax{\let\Mmapstochar\Mmapsto}
\CustomizeMathJax{\newcommand{\Longmmapsto}{\mathrel{\unicode{x027FE}}}}
\CustomizeMathJax{\newcommand{\Mmappedfrom}{\mathrel{\unicode{x02906}}}}
\CustomizeMathJax{\let\Mmappedfromchar\Mmappedfrom}
\CustomizeMathJax{\newcommand{\Longmmappedfrom}{\mathrel{\unicode{x027FD}}}}
\CustomizeMathJax{\let\Mmapsfrom\Mmappedfrom}% from kpfonts-otf
\CustomizeMathJax{\let\Longmmapsfrom\Longmmappedfrom}% from kpfonts-otf
%
\CustomizeMathJax{\newcommand{\boxright}{%
    \mathrel{\unicode{x025A1}\!\unicode{x02192}}%
}}
\CustomizeMathJax{\newcommand{\boxleft}{%
    \mathrel{\unicode{x02190}\!\unicode{x025A1}}%
}}
\CustomizeMathJax{\newcommand{\boxdotright}{%
    \mathrel{\unicode{x022A1}\!\unicode{x02192}}%
}}
\CustomizeMathJax{\newcommand{\boxdotleft}{%
    \mathrel{\unicode{x02190}\!\unicode{x022A1}}%
}}

```

```
\CustomizeMathJax{\newcommand{\Diamondright}{%
    \mathrel{\unicode{x025C7}\!\unicode{x02192}}%
}}
\CustomizeMathJax{\newcommand{\Diamondleft}{%
    \mathrel{\unicode{x02190}\!\unicode{x025C7}}%
}}
\CustomizeMathJax{\newcommand{\Diamonddotright}{%
    \mathrel{\unicode{x027D0}\!\unicode{x02192}}%
}}
\CustomizeMathJax{\newcommand{\Diamonddotleft}{%
    \mathrel{\unicode{x02190}\!\unicode{x027D0}}%
}}

\CustomizeMathJax{\newcommand{\boxRight}{%
    \mathrel{\unicode{x025A1}\!\unicode{x021D2}}%
}}
\CustomizeMathJax{\newcommand{\boxLeft}{%
    \mathrel{\unicode{x021D0}\!\unicode{x025A1}}%
}}
\CustomizeMathJax{\newcommand{\boxdotRight}{%
    \mathrel{\unicode{x022A1}\!\unicode{x021D2}}%
}}
\CustomizeMathJax{\newcommand{\boxdotLeft}{%
    \mathrel{\unicode{x021D0}\!\unicode{x022A1}}%
}}

\CustomizeMathJax{\newcommand{\DiamondRight}{%
    \mathrel{\unicode{x025C7}\!\unicode{x021D2}}%
}}
\CustomizeMathJax{\newcommand{\DiamondLeft}{%
    \mathrel{\unicode{x021D0}\!\unicode{x025C7}}%
}}
\CustomizeMathJax{\newcommand{\DiamonddotRight}{%
    \mathrel{\unicode{x027D0}\!\unicode{x021D2}}%
}}
\CustomizeMathJax{\newcommand{\DiamonddotLeft}{%
    \mathrel{\unicode{x021D0}\!\unicode{x027D0}}%
}}
\CustomizeMathJax{\newcommand{\Diamonddot}{\mathrel{\unicode{x027D0}}}}

\CustomizeMathJax{\newcommand{\circleright}{%
    \mathrel{\unicode{x025CB}\!\unicode{x02192}}%
}}
\CustomizeMathJax{\newcommand{\circleleft}{%
    \mathrel{\unicode{x02190}\!\unicode{x025CB}}%
}}
\CustomizeMathJax{\newcommand{\circledotright}{%
    \mathrel{\unicode{x02299}\!\unicode{x02192}}%
}}
\CustomizeMathJax{\newcommand{\circledotleft}{%
    \mathrel{\unicode{x02190}\!\unicode{x02299}}%
}}
\CustomizeMathJax{\let\circleddotright\circledotright}
\CustomizeMathJax{\let\circleddotleft\circledotleft}

\CustomizeMathJax{\newcommand{\multimapinv}{\mathrel{\unicode{x027DC}}}}
\CustomizeMathJax{\newcommand{\multimapboth}{\mathrel{\unicode{x029DF}}}}
\CustomizeMathJax{\newcommand{\multimapdot}{{\mathrel{-\!\bullet}}}}
\CustomizeMathJax{\newcommand{\multimapdotinv}{\mathrel{\bullet\!-}}}
\CustomizeMathJax{\newcommand{\multimapdotboth}{%
```

```
    \mathrel{{\bullet\!\!-\!\!\bullet}}%
}}
\CustomizeMathJax{\newcommand{\multimapdotbothA}{\mathrel{\unicode{x022B6}}}}
\CustomizeMathJax{\newcommand{\multimapdotbothB}{\mathrel{\unicode{x22B7}}}}

\CustomizeMathJax{\newcommand{\multimapbothvert}{%
    \mathrel{\overset{\unicode{x025CB}}{\underset{\unicode{x025CB}}{|}}}%
}}
\CustomizeMathJax{\newcommand{\multimapdotbothvert}{%
    \mathrel{\overset{\unicode{x025CF}}{\underset{\unicode{x025CF}}{|}}}%
}}
\CustomizeMathJax{\newcommand{\multimapdotbothBvert}{% bug in kpfonts-otf
    \mathrel{\overset{\unicode{x025CF}}{\underset{\unicode{x025CB}}{|}}}%
}}
\CustomizeMathJax{\newcommand{\multimapdotbothAvert}{% bug in kpfonts-otf
    \mathrel{\overset{\unicode{x025CB}}{\underset{\unicode{x025CF}}{|}}}%
}}

\CustomizeMathJax{\newcommand{\bignplus}{%
    \mathop{\LWRoverlaysymbols{\unicode{xFF0B}}{\unicode{x22C2}}}%
}}
\CustomizeMathJax{\let\bigcapplus\bignplus}
\CustomizeMathJax{\let\capplus\bignplus}% from kpfonts-otf

\CustomizeMathJax{\newcommand{\bigsqcapplus}{%
    \mathop{\LWRoverlaysymbols{\unicode{xFF0B}}{\unicode{x2A05}}}
}}
\CustomizeMathJax{\let\sqcapplus\bigsqcapplus}% from kpfonts-otf

\CustomizeMathJax{\newcommand{\bigsqcupplus}{%
    \mathop{\LWRoverlaysymbols{\unicode{xFF0B}}{\unicode{x2A06}}}
}}
\CustomizeMathJax{\let\sqcupplus\bigsqcupplus}% from kpfonts-otf

\CustomizeMathJax{\newcommand{\parallelslant}{\mathrel{\unicode{x02AFD}}}}
\CustomizeMathJax{\newcommand{\parallelbackslant}{%
    \mathrel{\unicode{x0005C}\!\!\unicode{x0005C}}%
}}

\CustomizeMathJax{\newcommand{\Eqqcolon}{\mathrel{=\!\unicode{x2237}}}}
\CustomizeMathJax{\let\eqqColon\Eqqcolon}% for kpfonts-otf
\CustomizeMathJax{\newcommand{\dashColon}{\mathrel{-\unicode{x2237}}}}
\CustomizeMathJax{\newcommand{\Colondash}{\mathrel{\unicode{x2237}-}}}

\CustomizeMathJax{\newcommand{\colonapprox}{\mathrel{:\approx}}}
\CustomizeMathJax{\newcommand{\colonsim}{\mathrel{:\sim}}}
\CustomizeMathJax{\newcommand{\Colonapprox}{%
    \mathrel{\unicode{x2237}\!\approx}%
}}
\CustomizeMathJax{\newcommand{\Colonsim}{\mathrel{\unicode{x2237}\!\sim}}}

\CustomizeMathJax{\newcommand{\strictif}{%
    \mathrel{\unicode{x0297D}}%
}}% right fish tail
\CustomizeMathJax{\newcommand{\strictfi}{%
    \mathrel{\unicode{x0297C}}%
}}% left fish tail
\CustomizeMathJax{\newcommand{\strictiff}{%
    \mathrel{\unicode{x0297C}\!\!\unicode{x0297D}}%
}}% left/right fish tails
```

```
\CustomizeMathJax{\newcommand{\circledwedge}{%
    \mathbin{\LWRoverlaysymbols{\unicode{x025EF}}{\unicode{x02227}}}%
}}
\CustomizeMathJax{\newcommand{\circledvee}{%
    \mathbin{\LWRoverlaysymbols{\unicode{x025EF}}{\unicode{0x02228}}}%
}}
\CustomizeMathJax{\newcommand{\circledbar}{\mathbin{\unicode{x029B6}}}}

\CustomizeMathJax{\newcommand{\openJoin}{%
    \mathrel{\unicode{x2AA4}}%
}}% overlapping ><
\CustomizeMathJax{\newcommand{\opentimes}{%
    \mathrel{\unicode{x2AA4}}%
}}% overlapping ><

\CustomizeMathJax{\newcommand{\VvDash}{\mathrel{\unicode{x22AA}}}}

\CustomizeMathJax{\newcommand{\lambdabar}{%
    \mathord{\LWRoverlaysymbols{\raise{.5ex}{-}}{\lambda}}%
}}

\CustomizeMathJax{\newcommand{\lambdaslash}{\mathord{\unicode{x019B}}}}

\CustomizeMathJax{\newcommand{\Wr}{%
    \mathbin{\unicode{x02240}\!\unicode{x02240}}%
}}

\CustomizeMathJax{\newcommand{\dashleftrightarrow}{%
    \mathrel{\unicode{x021E0}\!\unicode{x021E2}}%
}}
\CustomizeMathJax{\let\leftrightdasharrow\dashleftrightarrow}% for kpfonts-otf

\end{warpMathJax}
```

File 591 **lwarp-common-mathjax-overlaysymbols.sty**

§ 700 Package

common-mathjax-overlaysymbols

lwarp-common-mathjax-overlaysymbols (*Pkg*)

Common code used by a number of packages to overlay two symbols for MathJax.

for HTML output:

```
\ProvidesPackage{lwarp-common-mathjax-overlaysymbols}[2020/08/17]
```

\LWRoverlaysymbols {⟨*symbol*⟩} {⟨*symbol*⟩}

Overlays one symbol over another.

```
\begin{warpMathJax}

\CustomizeMathJax{\newcommand{\LWRoverlaysymbols}[2]{%
    \mathord{%
        \smash{%
            \mathop{#2\strut}%
            \limits^{\smash{\lower3ex{#1}}}%
        }%
        \strut%
    }%
```

```
}}

\end{warpMathJax}
```

Change History

§ 701 ## Chg Hist

For the most recent changes, see page 1334.

Index of Objects

This is an index of macros, environments, booleans, counters, lengths, packages, classes, options, keys, files, and various other programming objects. Each is listed by itself, and also by category. In some cases, they are further subdivided by [class].

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition.

C

D

E

G

H

M

N

O

P

Q

R

S

General Index

This is an index of instructions and concepts. Look here when wondering how to do something, and check the Troubleshooting Index when something goes wrong.

Troubleshooting Index

This index is a sorted reference of problems and solutions. In order to make it easier to locate a solution, the same issue may be addressed by more than one entry.

Entries starting with page 202 are often duplicates of entries with lower page numbers, as the same warning may occur within the user manual and again within the source code.

X

Index of Indexes