The biblatex-chicago package: Style files for biblatex

David Fussner   **Version 2.3a**
djf027@googlemail.com
November 17, 2022

Contents

1 Notice

This package is still under active development. The biblatex package by Philipp Lehman, Philip Kime, Moritz Wemheuer, Audrey Boruvka, and Joseph Wright is now quite stable, but my task of incorporating the many enhancements it has accumulated in recent releases is ongoing. The biblatex-chicago package itself now implements the 17th edition of the *Chicago Manual of Style*, though I have made it possible to continue to use the 16th edition files if that is imperative for you. The package relies heavily, in all styles, on using biber as its backend; other backends will not work properly.

I have tried to implement as much of the *Manual's* specification as possible, though undoubtedly some gaps remain. If it seems like this package could be of use to you, yet it doesn't do something you need/want it to do, please feel free to let me know, and of course any suggestions for solving problems more elegantly or accurately would be most welcome.

Important Note: If you have used biblatex-chicago before, especially if you've been using anything earlier than version 2.0, please make sure you have read the RELEASE file that came with the package. It details the changes you'll need to make to your .bib database in order for it to work properly with this release. If you have continued to use the 16th-edition styles, I do strongly recommend that you switch to the new edition, as I am deprecating the previous edition styles, and will delete them in a future release. Any required changes, as you can see from the RELEASE file, in the main involve *additions* to the specification, with required alterations to your existing .bib databases actually being rather rare, which should help ease the transition. If you are new to these styles or to biblatex itself, please do continue reading at least the following section.

2 Quickstart

The biblatex-chicago package is designed for writers who wish to use LaTeX and biblatex, and who either want or need to format their references according to one of the specifications defined by the *Chicago Manual of Style*. This package includes two versions of the *Manual's* "author-date" system, favored by many disciplines in the sciences and social sciences, and also its "notes & bibliography" style, generally favored in the humanities. The latter code produces a full reference in a first footnote, shorter references in subsequent notes, and a full reference in the bibliography. Some authors prefer to use the shorter note form even for the first occurrence, relying on the bibliography to provide the full information. This, too, is supported by the code. The author-date styles produce a short, in-text citation inside parentheses — (Author Year) — keyed to a list of references where entries start with the same name and year.

The documentation you are reading covers all three of these Chicago styles and their variants. I recommend that users new to the package read this Quickstart section first, perhaps then passing on to whichever of the two introductory files, cms-notes-intro.pdf or cms-dates-intro.pdf, is relevant to their needs, returning here afterward for more details on those parts of the functionality concerning which they still have questions. Much of what follows is relevant to all users, but I have decided, after some experimentation, to keep the instructions for the two author-date styles separate from those pertaining to the notes & bibliography style, at least in sections 4 and 5. Information provided under one style will often duplicate that found under the other, but efficiency's loss should, I hope, be clarity's gain, and much of what you learn using one style will be applicable without alteration to the other. Within the author-date section, the authordate-trad information really only appears separately in section 5.2, s.v. "title." Throughout the documentation, **New!** any green text indicates something new in this release, while **blue-green** text is a clickable link to an external document.

Here's a list of things you will need in order to use biblatex-chicago:

- The biblatex package, of course! The current version — 3.18b at the time of writing — has received extensive testing, and contains features and bug fixes upon which my code relies. Please don't use any earlier version. Biblatex requires several packages, and it strongly recommends several more:

    - biber — the next-generation BIBTEX replacement by Philip Kime and François Charette, available from SourceForge (required). You should use the latest version, 2.18, to work with biblatex 3.18b and biblatex-chicago; please note that any other backend will not produce accurate results.
    - e-TeX (required)
    - etoolbox — available from CTAN (required)
    - keyval — a standard package (required)
    - ifthen — a standard package (required)
    - url — a standard package (required). In biblatex-chicago.sty you'll see that I set `\urlstyle{rm}`. If you happen to use a non-roman text font, you may want to set `\urlstyle{same}` in your preamble instead.
    - babel — a standard package (*strongly* recommended)
    - csquotes — available from CTAN (recommended). Please upgrade to the latest version of csquotes (5.2n).

- The standard expl3 and xparse packages are loaded automatically for most users, and if they aren't biblatex-chicago does it for you.

- With recent changes both to biblatex and to biblatex-chicago, biblatex-chicago itself now requires two packages, which are both loaded for you if you load biblatex-chicago.sty, but which you'll have to load manually if not. They are:

    - nameref — a standard package, available in CTAN.
    - xstring — also standard and available in CTAN.

- The line:

      \usepackage[notes,backend=biber]{biblatex-chicago}

  in your document preamble to load the notes & bibliography style, the line:

      \usepackage[authordate,backend=biber]{biblatex-chicago}

  to load the author-date style, or the line:

      \usepackage[authordate-trad,backend=biber]{biblatex-chicago}

  to load the traditional variant of the author-date style. If you add "16" to any of the keys above, e.g.,

      \usepackage[authordate16,backend=biber]{biblatex-chicago}

  you can continue to use the 16th-edition styles, if that should prove necessary. Any other options you usually pass to biblatex can be given to biblatex-chicago instead, but loading it this way sets up a large number of other parameters automatically, parameters whose absence may surprise you when processing your documents. You can load the package via the usual \usepackage{biblatex}, adding either style=chicago-notes or style=chicago-authordate, but this is intended mainly for those, probably experienced users, who wish to set much of the low-level formatting of their documents themselves. Please see sections 4.5.1 and 5.5.1 below for a fuller discussion of the issues involved here, and please also remember to load xstring and nameref manually if you use this latter method.

- If you set \usepackage[notes,short,backend=biber]{biblatex-chicago} you'll get the short note format even in the first reference of a notes & bibliography document, letting the bibliography provide the full reference.

- If you are accustomed to using the natbib or mcite compatibility options with biblatex, then you can continue to do so with biblatex-chicago. If you are using \usepackage{biblatex-chicago} to load the style, then you can place (e.g.) natbib or natbib=true among its options to pass it through to biblatex. Please see sections 4.4.3 and 5.4.3, below.

- By far the simplest setup is to use babel, and to have american as the main text language. (Polyglossia should work, too, and has received somewhat more testing for this release.) As before, babel-less setups, and also those choosing english as the main text language, should work out of the box. Biblatex-chicago also provides (at least partial) support for Brazilian Portuguese, British, Dutch, Finnish, French, German, Icelandic, Norwegian, Romanian, Spanish, and Swedish. Please see below (section 7) for a fuller explanation of all the options.

- chicago-authordate.cbx, chicago-authordate-trad.cbx, chicago-dates-common.cbx, chicago-authordate.bbx, chicago-authordate-trad.bbx, chicago-notes.cbx, chicago-notes.bbx, cms-american.lbx, cms-brazilian.lbx, cms-british.lbx, cms-dutch.lbx, cms-finnish.lbx, cms-french.lbx, cms-german.lbx, cms-icelandic.lbx, cms-ngerman.lbx, cms-norsk.lbx, cms-norwegian.lbx, cms-nynorsk.lbx, cms-romanian.lbx, cms-spanish.lbx, cms-swedish.lbx, biblatex-chicago.sty, and cmsdocs.sty, all from biblatex-chicago, installed either in a system-wide TeX directory, or in the working directory where you keep your *.tex files. The .zip file from CTAN contains subdirectories to help keep the growing number of files organized, so the files listed above can be found in the latex/ subdirectory. If you install in a system-wide directory, I suggest a standard layout like <TEXMFLOCAL>/tex/latex/biblatex-contrib/biblatex-chicago, where <TEXMFLOCAL> is the root of your local TeX installation — for example, and depending on your operating system, something like /usr/share/texmflocal, /usr/local/share/texmf, or C:\Local TeX Files\. Then you can copy the contents of the latex/ directory there. (If you install into your working directory, then you'll need to copy the files directly there, without subdirectories.) Of course, if you've placed them anywhere in the texmf tree, you'll need to update the file name database to make sure TeX can find them.

- The files chicago-authordate16.cbx, chicago-authordate-trad16.cbx, chicago-dates-common16.cbx, chicago-authordate16.bbx, chicago-authordate-trad16.bbx, chicago-notes16.cbx, and chicago-notes16.bbx, which, as their names suggest, allow you to continue using the 16th-edition specifications alongside the most recent biblatex, if your project requires this. They can be found in the same directory as the 17th-edition equivalents.

- The dependent LaTeX package cmsendnotes.sty, found with the previous files. It offers additional functionality for those wishing to use the new noteref option with endnotes instead of footnotes. See section 4.4.4, below, and also cms-noteref-demo.pdf.

- The file cms.dbx, also to be found with the previous files, and a clone of the standard biblatex file 93-nameparts.dbx. It extends the default data model by adding new name parts, allowing the localized presentation of names from a number of linguistic and geographical contexts. Please see section 7.1 below for details of how to use it.

- The very clear and detailed documentation of the biblatex system, available in that package as biblatex.pdf. Here the authors explain why you might want to use the system, the rules for constructing .bib files for it, and the (numerous) methods at your disposal for modifying the formatted output.

- The files cms-notes-intro.pdf, cms-dates-intro.pdf, cms-trad-appendix.pdf, and cms-noteref-demo.pdf, the first two of which contain introductions to some of the main features of the Chicago styles, while the third documents some of the alterations you might need to make to your .bib files to use the trad style. The fourth gives a brief example of the usage of the noteref package option to the notes & bibliography style. All four are fully hyperlinked, the first three in particular allowing you easily to jump from notes or citations to an annotated bibliography or reference list, and thence to the .bib entries themselves. If you ensure that all four are in the same directory as the document you are reading (the TeX Live default), marginal links there will take you to further discussions here. The file cmsdocs.sty contains code and kludges designed specifically for compiling cms-dates-intro.tex, cms-notes-intro.tex and cms-trad-appendix.tex, so please *do not* load it yourself anywhere else, as it redefines and interferes with some of the macros from the main package.

- The annotated bibliography files notes-test.bib and dates-test.bib, and the not-yet-annotated legal-test.bib, all of which will acquaint you with many of the details on how to get started constructing your own .bib files for use with the three biblatex-chicago styles.

- The files cms-notes-sample.pdf, cms-dates-sample.pdf, cms-trad-sample.pdf, and cms-legal-sample.pdf. The first shows how my system processes notes-test.bib and cms-notes-sample.tex, in both footnotes and bibliography, the second and third are the result of processing dates-test.bib with cms-dates-sample.tex or cms-trad-sample.tex, and the fourth processes legal-test.bib using cms-legal-sample.tex. All of these files are in doc/, and the sample files, aside from the last named, are mainly included for testing purposes.

- The file you are reading, biblatex-chicago.pdf, which aims to be as complete a description as possible of the rules for creating a .bib file that will, when processed by LaTeX and biber, at least somewhat ease the burden when you try to implement the *Chicago Manual of Style*'s specifications. These docs may seem frustratingly overlong, but remember that you only need to read the part(s) that apply to the style in which you are interested. Much of the information in section 4 is duplicated in section 5, so even if you have a need for multiple styles then using one will be excellent preparation for the others. If you have used a previous version of this package, please pay particular attention to the sections on Obsolete and Deprecated Features, starting on page 185. You will find the nineteen previous files in the doc/ subdirectory once you've extracted biblatex-chicago.zip. If you wish to place them

in a system-wide directory, you can try: `<TEXMFLOCAL>/doc/latex/biblatex-contrib/biblatex-chicago`, all the while remembering, of course, to update the file name database afterward.

- Access to a copy of *The Chicago Manual of Style* itself, which naturally contains incomparably more information than I can hope to present here. It should always be your first port of call when any doubts arise as to exactly what the specification requires.

2.1 License

Copyright © 2008–2022 David Fussner. This package is author-maintained. This work may be copied, distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3 of this license or (at your option) any later version. The latest version of this license is in http://www.latex-project.org/lppl.txt and version 1.3 or later is part of all distributions of LaTeX version 2005/12/01 or later. This software is provided "as is," without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

2.2 Acknowledgements

Even a cursory glance at the cbx and bbx files in the package will demonstrate how much of biblatex's code I've adapted and re-used, and I've also followed some of the advice the authors have given to others in the comp.text.tex newsgroup and on Stackexchange. In particular, Philipp Lehman's advice on constructing biblatex-chicago.sty was invaluable. The code for formatting the footnote marks, and that for printing the separating rule only after a run-on note, I've adapted from the footmisc package by Robin Fairbairns, and I've borrowed ideas for the shorthandibid option from Dominik Waßenhoven's biblatex-dw package. I've adapted Audrey Boruvka's \textcite code from Stackexchange for the notes & bibliography style, and her page-number-compression code for both styles from the same site. The dependent package cmsendnotes.sty contains code by John Lavagnino and Ulrich Dirr. I am very grateful to Marçal Orteu Punsola for the Spanish localization, to Patrick Danilevici for the Romanian localization, to Wouter Lancee for the Dutch localization, to Gustavo Barros for the Brazilian Portuguese localization, to Stefan Björk for the Swedish localization, to Antti-Juhani Kaijahano for the Finnish localization, to Baldur Kristinsson for providing the Icelandic localization, and to Håkon Malmedal for the Norwegian localizations. Kazuo Teramoto and Gildas Hamel both sent patches to improve the package, and Arne Skjærholt provided some code to get me started on the \gentextcite commands. If there's other LaTeX code I've appropriated and forgotten, please remind me. Finally, Charles Schaum and Joseph Reagle Jr. were both extremely generous with their help and advice during the development of this package, and have both continued indefatigably to test it and suggest needed improvements. They were particularly instrumental in encouraging the greatest possible degree of compatibility with other biblatex styles. Indeed, if the task of adapting .bib files for use with the Chicago style seems onerous now, you should have tried it before they got their hands on it.

3 Detailed Introduction

The *Chicago Manual of Style*, implemented here in its 17th edition, has long, in America at least, been one of the most influential style guides for writers and publishers. While one's choices are now perhaps more extensive than ever, the *Manual* at least still provides a widely-recognized, and widely-utilized, standard. Indeed, when you add to this the sheer completeness of the specification, its detailed instructions for referencing an enormous number of different kinds of source material, then your choice (or your publisher's choice) of the *Manual* as a style guide seems set to be a happy one.

These very strengths, however, also make the style difficult to use. Admittedly, the *Manual* emphasizes consistency within a work, as opposed to rigid adherence to the specification, at least when writer and publisher agree (14.4). Sometimes a publisher demands such adherence, however, and anyone who has attempted to produce it may well come away with the impression that the specification itself is somewhat idiosyncratic in its

complexity, and I can't help but agree. In the notes & bibliography style, the numerous differences in punctuation (and strings identifying translators, editors, and the like) between footnotes and bibliographies and the sometimes unusual location of page numbers; in both styles the distinction between "journal" and "magazine," and the formatting differences between (e.g.) a work from antiquity and one from the Renaissance, all of these tend to overburden the writer who wants to comply with the standard. Many of these complexities, in truth, make the specification very nearly impossible to implement straightforwardly in a system like biblatex — options multiply, each requiring a particular sort of formatting, until one almost reaches the point of believing that every individual book or article should have its own entry type. Completeness and usability tend each to exclude the other, so the code you have before you is a first attempt to achieve the former without utterly sacrificing the latter.

What biblatex-chicago can and can't do

In short, the biblatex style files in this package try to simplify the task of following the two Chicago specifications along with their major variants. In the notes & bibliography style, the two sorts of reference are treated separately (as are the two different note forms, long and short), and you can choose always to use the short note form, even at the first citation. In the two author-date styles, a series of options allows you to choose which date (original printing, reprint, or both) appears in citations and at the head of entries in the list of references. In all styles, punctuation is placed within quotation marks when needed, and as a general rule as many parts of the style as possible are implemented as transparently as possible. Thanks to advice I received from Joseph Reagle Jr. and Charles Schaum while these files were a work in progress, I have attended as carefully as I can to backward compatibility with the standard biblatex styles, and have attempted to minimize both any changes you need to make to achieve compliance with the Chicago specification, and indeed also any changes necessary to switch between the two Chicago styles. There is no doubt room for improvement on this score, but even now, for a substantial number of entries, any well-constructed .bib file that works for other biblatex styles will "just work" under biblatex-chicago. By no means, however, will all entries in such a .bib file produce equally satisfactory results. Using this documentation and the examples in dates-test.bib and/or notes-test.bib, it should be possible to achieve compliance, though the amount of revision necessary to do so will vary significantly from .bib file to .bib file. Conversely, once you have created a database for biblatex-chicago, it won't necessarily work well with other biblatex styles. Indeed, most, quite possibly all, users will find that they need to use special formatting macros within the .bib file that would make such a file unusable in any other context. I strongly recommend, if you want to experiment with this style, that you work on a copy of any .bib files that are important to you, until you have determined that this package does what you need/want it to do.

When I first began working on this package, I made the decision to alter as little as possible the main files from Lehman's biblatex, so that my .bbx and .cbx files would use his original LaTeX .sty file and BibTeX .bst file. As you proceed, you will no doubt encounter some of the consequences of this decision, with certain fields and entry types in the .bib file having less-than-memorable names because I chose to use the supplementary ones provided by biblatex rather than alter that package's files. With additions to the standard data model now possible, this will be one of the directions for future development, particularly if other styles are adopting certain broad conventions. Needless to say, I'm open to advice and suggestions on this score.

4 The Specification: Notes & Bibliography

In what follows, I attempt to explain all the parts of biblatex-chicago-notes that might be considered somehow "non standard," at least with respect to the styles included with biblatex itself, though in the section on entry fields I have also duplicated a lot of the information in biblatex.pdf, which I hope won't badly annoy expert users of the system. New in this release Headings in green indicate either material new to this release or old material that has undergone significant revision. Numbers in parentheses refer to sections of the *Chicago Manual of Style*, 17th edition. The file notes-test.bib contains many examples from the *Manual*

which, when processed using biblatex-chicago-notes, should produce the same output as you see in the *Manual* itself, or at least compliant output, where the specifications are vague or open to interpretation, a state of affairs which does sometimes occur. I have provided cms-notes-sample.pdf, which shows how my system processes notes-test.bib, and I have also included the reference keys from the latter file below in parentheses.

4.1 Entry Types

The complete list of entry types currently available in biblatex-chicago-notes, minus the odd biblatex alias, is as follows: **article**, **artwork**, **audio**, **book**, **bookinbook**, **booklet**, **collection**, **customc**, **dataset**, **image**, **inbook**, **incollection**, **inproceedings**, **inreference**, **jurisdiction**, **legal**, **legislation**, **letter**, **manual**, **misc**, **music**, **mvbook**, **mvcollection**, **mvproceedings**, **mvreference**, **online** (with its alias **www**), **patent**, **performance**, **periodical**, **proceedings**, **reference**, **report** (with its alias **techreport**), **review**, **standard**, **suppbook**, **suppcollection**, **suppperiodical**, **thesis** (with its aliases **mastersthesis** and **phdthesis**), **unpublished**, and **video**.

What follows is an attempt to specify all the differences between these types and the standard provided by biblatex. If an entry type isn't discussed here, then it is safe to assume that it works as it does in the standard styles. In general, I have attempted not to discuss specific entry fields here, unless such a field is crucial to the overall operation of a given entry type. As a general and important rule, most entry types require very few fields when you use biblatex-chicago-notes, so it seemed to me better to gather information pertaining to fields in the next section.

article The *Chicago Manual of Style* (14.164) recognizes three different sorts of periodical publication, "journals," "magazines," and "newspapers." The first (14.166) is "a scholarly or professional periodical available mainly by subscription," while the second refers to "weekly or monthly (or sometimes daily)" publications that are "available in individual issues at libraries or bookstores or newsstands or offered online, with or without a subscription." "Magazines" will tend to be "more accessible to general readers," and typically won't have a volume number. Indeed, by fiat I declare that should you need to refer to a journal that identifies its issues mainly by year, month, or week, then for the purposes of biblatex-chicago-notes such a publication is a "magazine," and not a "journal."

For articles in "journals" you can simply use the traditional BIBTEX — and indeed biblatex — article entry type, which will work as expected and set off the page numbers with a colon, as required by the *Manual*. If, however, you need to refer to a "magazine" or a "newspaper," then you need to add an entrysubtype field containing the exact string `magazine` or, now, its synonym `newspaper`. The main formatting differences between a `magazine`/`newspaper` and a plain article are that the year isn't placed within parentheses, and that page numbers are set off by a comma rather than a colon. Otherwise, the two sorts of reference have much in common. (For article, see *Manual* 14.168–87; batson, beattie:crime, friedman:learning, garaud:gatine, garrett, hlatky:hrt, kern, lewis, loften:hamlet, mcmillen:antebellum, rozner:liberation, saberhagen:beluga, warr:ellison, white:callimachus. For entrysubtype `magazine`, see 14.171, 14.188–200; assocpress:gun, morgenson:market, reaves:rosen, stenger:privacy.)

It gets worse. The *Manual* treats reviews (of books, plays, performances, etc.) as a sort of recognizable subset of "journals," "magazines," and "newspapers," distinguished mainly by the way one formats the title of the review itself. Biblatex provides a review entry type which will handle a large subset of such citations, though not all. The key rule is this: if a review has a separate, non-generic title (gibbard; osborne:poison) in addition to something that reads like "review of ...," then you need an article entry, with or without the `magazine` entrysubtype, depending on the sort of publication containing the review. If the only title is the generic "review of ...," for example, then you'll need the review entry type, with or without this same entrysubtype toggle using `magazine`. On review entries, see below.

In the case of a review with a specific as well as a generic title, the former goes in the title field, and the latter in the titleaddon field. Standard biblatex intends this field for use with additions to titles that may need to be formatted differently from the titles themselves,

7

and biblatex-chicago-notes uses it in just this way, with the additional wrinkle that it can, if needed, replace the title entirely, and this in, effectively, any entry type, providing a fairly powerful, if somewhat complicated, tool for getting biblatex to do what you want. Here, however, if all you need is a titleaddon, then you want to switch to the review type, where you can simply use the title field instead.

Biblatex-chicago also, at the behest of Bertold Schweitzer, supports the relatedtype reviewof, which allows you to use the related mechanism to provide information about the work being reviewed, thereby simplifying how much information you need to provide in the reviewing entry. In particular, it relieves you of the need to construct titleaddon or title fields like: review of \mkbibemph{Book Title} by Author Name, as the related entry's title automatically provides the titleaddon in the article type and the title in the review type, with the related mechanism providing the connecting string. This may be particularly helpful if you need to cite multiple reviews of the same work; please see section 4.2.1 for further details.

No less than ten more things need explication here. First, since the *Manual* specifies that much of what goes into a titleaddon field stays unformatted — no italics, no quotation marks — this plain style is the default for such text, which means that you'll have to format any titles within titleaddon yourself, e.g., with \mkbibemph{}. (The related mechanism just mentioned provides this automatically.) Second, the *Manual* specifies a similar plain style for the titles of other sorts of material found in "magazines" and "newspapers," e.g., obituaries, letters to the editor, interviews, the names of regular columns, and the like. References may contain both the title of an individual article and the name of the regular column, in which case the former should go, as usual, in a title field, and the latter in titleaddon. As with reviews proper, if there is only the generic title, then you want the review entry type. (See 14.191, 14.195–96; morgenson:market, reaves:rosen.)

Third, the *Manual* has slightly complicated instructions concerning "unsigned newspaper articles or features" (14.199). First, it suggests that such pieces are "best dealt with in text or notes." If, however, "a bibliography entry should be needed, the title of the newspaper stands in place of the author." The examples it provides, therefore, suggest quite different treatments of the same material in notes and bibliography, and they don't at any point that I can see recommend a format for short notes. I've implemented these recommendations fairly literally, which means that in an article entry, entrysubtype magazine, or in a review entry, entrysubtype magazine, and *only* in such entries, a missing author field results in the name of the periodical (in the journaltitle field) being used as the missing author, but *only* in the bibliography and in short notes. In long notes, the title will appear first, before the journaltitle. Note that the use of the name of the newspaper as an author creates sorting issues in the bibliography, issues that will mostly be solved for you if you use Biber as the backend. If you don't, or if the journaltitle begins with a definite or indefinite article with which you can't dispense, then you'll need a sortkey field to ensure that the bibliography entry is alphabetized correctly. (See lakeforester:pushcarts and, for the sorting issue, \DeclareSortingTemplate in section 4.4.1 below.)

Fourth, Bertold Schweitzer has pointed out, following the *Manual* (14.183), that while an issuetitle often has an editor, it is not too unusual for a title to have, e.g., an editor and/or a translator. In order to allow as many permutations as possible on this theme, I have brought the article entry type into line with most of the other types in allowing the use of the namea and nameb fields in order to associate an editor or a translator specifically with the title. The editor and translator fields, in strict homology with other entry types, are associated with the issuetitle if one is present, and with the title otherwise. The usual string concatenation rules still apply — cf. editor and editortype in section 4.2, below.

Fifth, in certain fields just beginning your data with a lowercase letter activates the mechanism for capitalizing that letter depending on its context within a note or bibliography entry. Please see \autocap in section 4.3.1 below for the details, but both the titleaddon and note fields are among those treating their data this way, and since both appear regularly in article entries, I thought the problem merited a mention here.

Sixth, if you need to cite an entire issue of any sort of periodical, rather than one article in an issue, then the periodical entry type, once again with or without the magazine

toggle in entrysubtype, is what you'll need. (You can also use the article type, placing what would normally be the issuetitle in the title field and retaining the usual journaltitle field, but this arrangement isn't compatible with standard biblatex.) The note field is where you place something like "special issue" or "supplement" (with the small "s" enabling the automatic capitalization routines), whether you are citing one article or the whole issue (14.178–79; conley:fifthgrade, good:wholeissue). Indeed, this is a somewhat specialized use of note, and if you have other sorts of information you need to include in an article, periodical, or review entry, then you shouldn't put it in the note field, but rather in titleaddon or perhaps addendum (brown:bremer).

Seventh, if you wish to cite certain kinds of television or radio broadcast, most notably interviews but perhaps also news segments or other "journalistic" material, then the article type, entrysubtype magazine is the place for it. The name of the program as a whole would go in journaltitle, with the name of the episode or segment in title, and the network's name in the usera field. Of course, if the piece you are citing has only a generic name (an interview, for example), then the review type would be the best place for it (8.189, 14.213; see bundy:macneil for an example of how this all might look in a .bib file.) Other sorts of broadcast, usually accessible through commercial recordings, would need one of the audiovisual entry types, probably audio (danforth:podcast) or video (friends:leia), while recordings from archives fit best either into online or into misc entries with an entrysubtype (coolidge:speech, roosevelt:speech).

Eighth, the *Manual* (14.208) specifies that blogs and other, similar online material should be presented like articles, with magazine entrysubtype (ellis:blog), and needn't appear in a bibliography. The title of the specific entry goes in title, the general title of the blog goes in journaltitle, and the word "blog" in the location field (though you could just use special formatting in the journaltitle field itself, which may sometimes be necessary). The 17th edition specifies that "blogs that are part of a larger publication should include the name of that publication." This usually involves a newspaper or magazine which also publishes various blogs on its website, and it means that such entries need a more general title than the journaltitle. It's not standard biblatex or anything, but you can now put such information in maintitle (with mainsubtitle and maintitleaddon, if needed), but only in article and review entries with a magazine entrysubtype (amlen:hoot). To cite a whole blog, you'll need the periodical entry type, with a title instead of a journaltitle, along with a (possible) maintitle. Comments on blogs, with generic titles like "comment on" or "reply to," need a review entry with the same entrysubtype. Such comments make particular use of the eventdate and of the nameaddon fields; please see the documentation of **review**, below, and also of the relatedtype commenton in section 4.2.1.

Ninth, the special biblatex field shortjournal allows you to present shortened journaltitles in article, review, and periodical entries, as well as facilitating the creation of lists of journal abbreviations in the manner of a shorthand list. Please see the documentation of **shortjournal** in section 4.2 for all the details on how this works.

Finally, the 17th edition (14.191) specifies that, for news sites carrying "stories as they unfold, it may be appropriate to include a time stamp for an article that includes one." You can provide this by using the standard biblatex time stamp format inside the date field, e.g., 2008-07-01T10:18:00. Since the *Manual* prefers the standard time zone initialisms, a separate timezone field would be required if you want to provide one.

If you're still with me, allow me to recommend that you browse through notes-test.bib to get a feel for just how many of the *Manual*'s complexities the article and review (and, indeed, periodical) types attempt to address. It may be that in future releases of biblatex-chicago-notes I'll be able to simplify these procedures somewhat, but in the meantime it might be of some comfort that I have found in my own research that the unusual and/or limit cases are really rather rare, and that the vast majority of sources won't require any knowledge of these onerous details.

artwork Arne Kjell Vikhagen pointed out to me that none of the standard entry types were straightforwardly adaptable when referring to visual artworks. It's unclear whether the *Manual* (14.235) believes it necessary to include them in the bibliographical apparatus at all, but it's easy all the same to conceive of contexts in which a list of artworks studied might

be desirable, and `biblatex` includes entry types for just this purpose, though the standard styles leave them undefined. `Biblatex-chicago` defines both artwork and image, which are in fact now clones of each other, so you can use either of them indifferently, the distinction existing only for historical reasons.

Constructing an entry is fairly straightforward. As one might expect, the artist goes in author and the name of the work in title. The type field is intended for the medium — e.g., oil on canvas, charcoal on paper, albumen print — and the version field might contain the state of an etching. You can place the dimensions of the work in note, and the current location in organization, institution, and/or location, in ascending order of generality. The type field, as in several other entry types, uses biblatex's automatic capitalization routines, so if the first word only needs a capital letter at the beginning of a sentence, use lowercase in the .bib file and let biblatex handle it for you. (See *Manual* 3.22, 8.198; leo:madonna, bedford:photo.)

The 17th edition of the *Manual* has included new information in some of its examples, so I have added 4 new fields to the drivers. Alongside the usual date for the creation of a work, you may also want to include the printing date of a particular exemplar of a photograph or a print. The system I have designed uses the *earlier* of the date and the origdate to be the date of creation, and the *later* to be the printing date. The style will automatically prefix the printing date with the localized `\bibstring printed`, so if that's the wrong string entirely then you can define userd any way you like to change it. If only *one* of those two dates is available, it will always serve as a creation date.

One of the *Manual's* examples is of a photograph published in a periodical, and information about this publication appears late in the entry, after the type. I have included the howpublished field so that you can give information about the periodical (meaning that you'll have to format the title yourself with `\mkbibemph`), and the eventdate field for you to provide the date of publication (mccurry:afghangirl).

As a final complication, the *Manual* (8.198) says that "the names of works of antiquity ... are usually set in roman." If you should need to include such a work in the reference apparatus, you can either define an entrysubtype for an artwork entry — anything will do — or you could use the misc entry type with an entrysubtype. Assuming the complicated date handling I've just outlined isn't required for such a work, in this instance the other fields in a misc entry function pretty much as in artwork.

audio Following the request of Johan Nordstrom, I have included three entry types, all undefined by the standard styles, designed to allow users to present audiovisual sources in accordance with the Chicago specifications. The *Manual's* presentation of such sources (14.261–68), though admirably brief, seems to me somewhat inconsistent, though perhaps I'm merely unable to spot the important regularities. The proliferation of online sources has made the task yet more complex, requiring the inclusion of the **article**, the **online**, and even the **misc** entry types, which see, under the audiovisual rubric. I shall attempt to delineate the main differences here, and though there are likely to be occasions when your choice of entry type is not obvious, at the very least biblatex-chicago should help you maintain consistency.

The **music** type is intended for all musical recordings that do not have a video component. This means, for example, digital media (whether on CD or hard drive), vinyl records, and tapes. The **video** type includes most visual media, whether it be films, TV shows, tapes and DVDs of the preceding or of any sort of performance (including music), or online multimedia. The *Manual's* treatment (14.267) of the latter suggests that online video excerpts, short pieces, and interviews should generally use the **online** type (horowitz:youtube, pollan:plant) or the **article** type (harwood:biden, kessler:nyt), depending on whether the pieces originate from an identifiably "journalistic" outlet. The **audio** type, our current concern, fills gaps in the others, and presents its sources in a more "book-like" manner. Published musical scores need this type — unpublished ones would use misc with an entrysubtype (shapey:partita) — as do podcasts and such favorite educational formats as the slideshow and the filmstrip (danforth:podcast, greek:filmstrip, schubert:muellerin, verdi:corsaro). The *Manual* (14.264) sometimes uses a similar format for audio books (twain:audio), though, depending on the sorts of publication facts you wish to present,

this sort of material may fall under music (auden:reading). Dated audio recordings that are part of an archive, online or no, may be presented either in an **online** or in a **misc** entry with an entrysubtype, the difference mainly being in just how closely associated the date will be with the title (coolidge:speech, roosevelt:speech). Actual radio broadcasts (as opposed to podcasts) pose something of a conundrum. Interviews and other sorts of "journalistic" material fit well into article or review entries (14.213), but other sorts of broadcast are not well represented in the *Manual's* examples (8.189), and what little there is suggests that, counter-intuitively, the video type is the best fit, as it is well equipped to present broadcasts of any sort.

Once you've accepted the analogy of composer to author, constructing an audio entry should be fairly straightforward, since many of the fields function just as they do in book or inbook entries. Indeed, please note that I compare it to both these other types as, in common with the other audiovisual types, audio has to do double duty as an analogue for both books and collections, so while there will normally be an author, a title, a publisher, a date, and a location, there may also be a booktitle and/or a maintitle — see schubert:muellerin for an entry that uses all three in citing one song from a cycle. (As with the music and video types, you can cite an individual piece separate from any large collection by using the title field and by defining an entrysubtype, which will stop biblatex-chicago italicizing your title in the absence of a booktitle.) If the medium in question needs specifying, the type field is the place for it. Please note, also, that while the titleaddon field can still specify creative or editorial functions for which biblatex-chicago provides no automated, localized handling, you can also now provide the string you need in an editor[abc]type field, e.g., "libretto by" (verdi:corsaro).

For podcasts, newly covered by the 17th edition (14.267), the audio type provides the nearest analogue I could find, and in general most of the data should fit comfortably into the fields already discussed above, the episode name in title and the name of the podcast in booktitle, for starters. Two details, however, need mentioning: first, the note field as the place to specify that it is a podcast, and the eventdate field for the date of publication of the specific episode (title) cited, which appears in close association with that title. Indeed, the eventdate field helps biblatex-chicago know that the audio entry is a podcast episode, and helps it construct the entry appropriately (danforth:podcast).

book   This is the standard biblatex and BIBTEX entry type, and the package can automatically provide abbreviated references in notes and bibliography when you use a crossref or an xref field. The functionality is not enabled by default, but you can enable it in the preamble or in the options field using the booklongxref option. Please see **crossref** in section 4.2 and booklongxref in section 4.4.2, below. Cf. harley:ancient:cart, harley:cartography, and harley:hoc for how this might look. The book type is useful also to present multimedia app content, the added fields version and type providing information about the app's version and about the system on which it runs (14.268; angry:birds).

bookinbook   This type provides the means of referring to parts of books that are considered, in other contexts, themselves to be books, rather than chapters, essays, or articles. Such an entry can have a title and a booktitle, but it can also contain a maintitle, all three of which will be italicized when printed. In general usage it is, therefore, rather like the traditional inbook type, only with its title in italics rather than in quotation marks. As with the book type, you can automatically enable abbreviated references in notes and bibliography, though this isn't the default. Please see **crossref** in section 4.2 and booklongxref in section 4.4.2, below. (Cf. *Manual* 14.109, 14.122, 14.124; bernhard:boris, bernhard:ritter, and bernhard:themacher for the abbreviating functionality; also euripides:orestes [treated differently in 14.122 and 14.124], plato:republic:gr.)

booklet   This is the first of two entry types — the other being manual, on which see below — which are traditional in BIBTEX styles, but which the *Manual* (14.220) suggests may well be treated basically as books. In the interests of backward compatibility, biblatex-chicago-notes will so format such an entry, which uses the howpublished field instead of a standard publisher, though of course if you do decide just to use a book entry then any information you might have given in a howpublished field should instead go in publisher. (See clark:mesopot.)

collection   This is the standard biblatex entry type, but the package can automatically provide abbreviated references in notes and bibliography when you use a crossref or an xref field. The functionality is not enabled by default, but you can enable it in the preamble or in the options field using the new booklongxref option. Please see **crossref** in section 4.2 and booklongxref in section 4.4.2, below. See harley:ancient:cart, harley:cartography, and harley:hoc for how this might look.

customc   This entry type allows you to include alphabetized cross-references to other, separate entries in the bibliography, particularly to other names or pseudonyms, as recommended by the *Manual*. (This is different from the crossref, xref, userf and related mechanisms, all primarily designed to include cross-references to other works. Cf. 14.81–82). The lecarre:cornwell entry, for example, would allow your readers to find the more-commonly-used pseudonym "John Le Carré" even if they were, for some reason, looking under his real name "David John Moore Cornwell." As I read the specification, these cross-references are particularly encouraged, bordering on required, when "a bibliography includes two or more works published by the same author but under different pseudonyms." The following entries in notes-test.bib show one way of addressing this: creasey:ashe:blast, creasey:york:death, creasey:morton:hide, ashe:creasey, york:creasey and morton:creasey.

In these latter cases, you would need merely to place the pseudonym in the author field, and the author's real name, under which their works are presented in the bibliography, in the title field. To make sure the cross-reference also appears in the bibliography, you can either manually include the entry key in a \nocite command, or you can put that entry key in the **userc** field in the work's main .bib entry, in which case biblatex-chicago will print the cross-reference if and only if you cite the main work. (Cf. userc, below.)

Under ordinary circumstances, biblatex-chicago will connect the two parts of the cross-reference with the word "*See*" — or its equivalent in the document's language — in italics. If you wish to present it differently, you can put the connecting word(s) into the nameaddon field, formatted as you wish.

dataset   This entry type, new in biblatex 3.13, allows you to cite scientific databases, for which the *Manual* (14.257) presents some rather specific, if brief, instructions. To construct your entry, you can put the name of the database into author, a "descriptive phrase or record locator" in the title field, and if there's a specific accession number needed beyond the record locator you can put it into the number field, with the type field reserved to help explain what sort of number is involved. The howpublished field can also be used to provide extra descriptive detail about the number, if needed. More generally, a url will locate the database as a whole and a urldate will specify the date you accessed it. If, for some reason, an additional date is relevant, then the date field is available, while the pubstate field will appear before the date in case you need to modify the latter. (See 14.257; genbank:db, nasa:db.)

image   This entry type is now a clone of the artwork type, which see. I retain it here for historical reasons (See 3.22, 8.198; bedford:photo.)

inbook   These two standard biblatex types have very nearly identical formatting requirements
incollection   as far as the Chicago specification is concerned, but I have retained both of them for compatibility. Biblatex.pdf (§ 2.1.1) intends the first for "a part of a book which forms a self-contained unit with its own title," while the second would hold "a contribution to a collection which forms a self-contained unit with a distinct author and its own title." The title of both sorts will be placed within quotation marks, and in general you can use either type for most material falling into these categories. I have, in both types, implemented the *Manual's* recommendations for space-saving abbreviations in notes and bibliography when you cite multiple pieces from the same collection. These abbreviations are activated by default when you use the crossref or xref field in incollection entries and in inbook entries, because although the *Manual* (14.108) here specifies a "multiauthor book," I believe the distinction between the two is fine enough to encourage similar treatments. (For more on this mechanism see **crossref** in section 4.2, below, and

the new option `longcrossref` in section [4.4.2]. Please note that it is also active by default in letter and inproceedings entries.) If the part of a book to which you are referring has had a separate publishing history as a book in its own right, then you may wish to use the bookinbook type, instead, on which see above. (See *Manual* 14.106–9; inbook: ashbrook:brain, phibbs:diary, will:cohere; incollection: centinel:letters, contrib:contrib, sirosh:visualcortex; ellet:galena, keating:dearborn, and lippincott:chicago [and the collection entry prairie:state] demonstrate the use of the crossref field with its attendant abbreviations in notes and bibliography.)

NB: The *Manual* suggests that, when referring to a chapter, one use either a chapter number or the inclusive page numbers, not both. If, however, you wish to refer in a footnote to a specific page within the chapter, biblatex-chicago-notes will always print the optional, postnote argument of a \cite command — the page number, say — instead of any inclusive page numbers given in the .bib file incollection entry. This mechanism is quite general, that is, any specific page reference given in any sort of \cite command overrides the contents of a pages field in a .bib file entry.

inproceedings This entry type works pretty much as in standard biblatex, even more so now that, after a request from Patrick Danilevici, I have included the eventdate, eventtitle, eventtitleaddon, and venue fields for specifying where and when the event occurred that produced the proceedings. These four fields are the main difference between it and incollection, along with the lack of an edition field and the possibility that an organization may be cited alongside the publisher, even though the *Manual* doesn't specify the use of any of these supplementary fields (14.217). Please note, also, that the crossref and xref mechanism for shortening citations of multiple pieces from the same proceedings is operative here, just as it is in incollection and inbook entries. See **crossref** in section [4.2] and the option `longcrossref` in section [4.4.2] for more details.

inreference This entry type is aliased to incollection in the standard styles, but the *Manual* has particular requirements, so if you are citing "[w]ell-known reference books, such as major dictionaries and encyclopedias," then this type should simplify the task of conforming to the specifications (14.232–34). The main thing to keep in mind is that I have designed this entry type for "alphabetically arranged" works, which you shouldn't cite by page, but rather by the name(s) of the article(s). Because of the formatting required by the *Manual*, we need one of biblatex's list fields for this purpose, and in order to keep all this out of the way of the standard styles, I have chosen the lista field. You should present these article names just as they appear in the work, separated by the keyword "`and`" if there is more than one, and biblatex-chicago-notes will provide the appropriate prefatory string (`s.v.`, plural `s.vv.`), and enclose each in its own set of quotation marks (ency:britannica). In a typical inreference entry very few other fields are needed, but "if a physical edition is cited, not only the edition number (if not the first) but also the date the volume or set was issued must be specified." In practice, this means a title, date, and possibly an edition field.

There are quite a few other peculiarities to explain here. First of all, you should present any well-known works *only* in notes, not in a bibliography, as your readers are assumed to know where to go for such a reference. You can use the `skipbib` option to achieve this. For such works, and given how little information will be present even in a full note, you may wish to use \fullcite or \footfullcite in place of the short form, especially if, for example, you are citing different versions of an article appearing in different editions.

If the work is slightly less well known, it may be that full publication details are appropriate (times:guide). Also, you can put an article name in the postnote field of inreference entries and have it properly formatted for you, and this holds for both long and short notes, which could allow you to refer separately to many different articles from the same reference work using only one .bib entry. (In a long note, any postnote field stops the printing of the contents of lista.) The only limitation on this system is that the postnote field, unlike lista, is not a biblatex list, and therefore for the formatting to work correctly you can only put one article name in it. Despite this limitation, I hope that the system might simplify things for users who cite numerous works of reference.

If it seems appropriate to include such a work in the bibliography, be aware that the contents of the lista field will also be presented there, which may not be what you want. A separate reference entry might well solve this problem. (The sorting issues with inreference, mvreference, and reference entries should no longer exist, as they should now always sort by title rather than by any author, editor, or namec that might also be present. If the title starts with a definite or indefinite article then a sortkey may still be necessary.)

Speaking of the author, this field holds the author of the specific entry (in lista), not the author of the title as a whole. This name will be printed after the entry's name (grove:sibelius). If you wish to refer to a reference work by author or indeed by editor, having either appear at the head of the note (long or short) or bibliography entry, then you'll need to use a book entry instead (cf. schellinger:novel), where the lista mechanism will also work in the bibliography, but which in every other way will be treated as a normal book, often a good choice for unfamiliar or non-standard reference works.

Finally, all of these rules apply to online reference works, along with a few more. The 17th edition of the *Manual* now allows, "subject to editorial discretion," the alternative treatment of an online reference work which "does not have (and never had) a printed counterpart" (14.206, 14.233). In effect this means that it can be treated more like an online entry than a book, its title therefore in plain roman rather than in italics. You can achieve this in inreference entries by providing an entrysubtype in the entry. Online reference works need not only a url but also, always, a urldate (instead of a date), as these sources are in constant flux. When that flux is of a particularly high frequency, as with Wikipedia, then a time stamp may also be needed. You can provide this in the urldate urlstamp=true field itself, using the standard biblatex format, e.g., 2008-07-01T10:18:00. It is possible to turn off the printing of the urltime with the urlstamp option, which is set to true by default, but which can be changed in your preamble for the whole document, for specific entry types, or in the options field of specific entries (wikiped:bibtex, grove:sibelius).

jurisdiction I document these three types in section 6 below, both because they all follow the speci-
legal fications of the *Bluebook* instead of the *Manual*, and also because they are the only entry
legislation types treated identically by the notes & bibliography style and the author-date styles.

letter This is the entry type to use for citing letters, memoranda, or similar texts, but *only* when they appear in a published collection. (Unpublished material of this nature needs a misc entry, for which see below.) Depending on what sort of information you need to present in a citation, you may simply be able to get away with a standard book entry, which may then be cited by page number (see 14.104; meredith:letters, adorno:benj). If, however, for whatever reason, you need to give full details of a specific letter, then you'll need to use the letter entry type, which attempts to simplify for you the *Manual*'s rather complicated rules for formatting such references. (See 14.111; jackson:paulina:letter, white:ross:memo, white:russ [a completely fictitious entry to show the crossref mechanism], white:total [a book entry, for the bibliography]).

To start, the name of the letter writer goes in the author field, while the title field contains both the name of the writer and that of the recipient, in the form `Author to Recipient`. The titleaddon field contains, optionally, the type of correspondence involved. If it's a letter, the type needn't be given, but if it's a memorandum or report or the like, then this is the place to specify that fact. Also, because the origdate field only accepts numbers, if you want to use the abbreviation "n.d." (or `\bibstring{nodate}`) for undated letters, then this is where you should put it. If you need to specify where a letter was written, then you can also use this field, and, if both are present, remember to separate the location from the type with a comma, like so: `memorandum, London`. Alternatively, you can put the place of writing into the origlocation field. Most importantly, the date of the letter itself goes in the origdate field (`year-month-day`), which allows a full date specification, while the publishing date of the whole collection goes in the date field. As in other entry types, then, the date field has its ordinary meaning of "date of publication." (You may have noticed that the presentation of the origdate in this sort of reference uses the day-month-year format, unlike the month-day-year format seen elsewhere. This follows a suggestion that material with "many references to specific dates" may use this format [6.38, 9.35, 14.224]. I should, I guess, make this configurable.) Another difficulty arises

when producing the short footnote form, which requires you to provide a shorttitle field of the form "to Recipient," the letter name as short as possible while avoiding ambiguity. The \letterdatelong command can be used in the postnote field of the citation to print the origdate, a possible aid to disambiguation. The remaining fields are fairly self explanatory, but do remember that the title of the published collection belongs in booktitle rather than in title.

Finally, the *Manual* specifies that if you cite more than one letter from a given published collection, then the bibliography should contain only a reference to said collection, rather than to each individual letter, while the form of footnotes would remain the same. This should be possible using BIBTEX's and biblatex's standard crossref field, with each letter entry pointing to a collection or book entry, for example. (If you are using Biber, then letter entries correctly inherit fields from book and collection entries, and also from the mvbook and mvcollection types — titles from the former provide a booktitle and from the latter a maintitle.) I shall discuss cross references at length later (see esp. **crossref** in section 4.2, below), but I should mention here that letter is one of the entry types in which a crossref or an xref field automatically results in special shortened forms in notes and bibliography if more than one piece from a single collection is cited. (The other entry types are inbook, incollection, and inproceedings; see 14.108 for the *Manual*'s specification.) This ordinarily won't be an issue for letter entries in the bibliography, as individual letters aren't included there, but it is operative in notes, where you can disable it by setting the longcrossref=true option, on which see section 4.4.2, below. To stop individual letters turning up in the bibliography, you can use the skipbib option in the options field.

manual   This is the second of two traditional BIBTEX entry types that the *Manual* suggests formatting as books, the other being booklet. As with this latter, I have retained it in biblatex-chicago-notes for backward compatibility, its main peculiarity being that, in the absence of a named author, the organization producing the manual will be printed both as author and as publisher. If you are using Biber you no longer need a sortkey field to aid biblatex's alphabetization routines, as the style takes care of this for you (cf. section 4.4.1, below). You also don't need to provide a shortauthor field, as the style will automatically use organization in the absence of anything else. Of course, if you were to use a book entry for such a reference, then you would need to define both author and publisher using the name you here might have put in organization. (See 14.84; chicago:manual, dyna:browser, natrecoff:camera. Cp. also the **standard** entry type.)

misc   As its name suggests, the misc entry type was designed as a hold-all for citations that didn't quite fit into other categories. In biblatex-chicago-notes, I have somewhat extended its applicability, while retaining its traditional use. Put simply, with no entry-subtype field, a misc entry will retain backward compatibility with the standard styles, so the usual howpublished, version, and type fields are all available for specifying an otherwise unclassifiable text, and the title will be italicized. (The *Manual*, you may wish to note, doesn't give specific instructions on how such citations should be formatted, so when using the Chicago style I would recommend you have recourse to this traditional entry type as sparingly as possible.)

If you do provide an entrysubtype field, the misc type provides a means for citing unpublished letters, memoranda, private contracts, wills, interviews, and the like, making it something of an unpublished analogue to the letter, article, and review entry types (which see). It also works well for presenting online audio pieces, particularly dated ones, like speeches. Typically, such an entry will cite part of an archive, and equally typically the text cited won't have a specific title, but only a generic one, whereas an unpublished entry will ordinarily have a specific author and title, and won't come from a named archive. The misc type with an entrysubtype defined is the least formatted of all those specified by the *Manual*, so titles are in plain text, and any location details take no parentheses in full footnotes. (It is quite possible, though somewhat unusual, for archival material to have a specific title, rather than a generic one. In these cases, you will need to enclose the title inside a \mkbibquote command manually. Cf. roosevelt:speech, shapey:partita.)

If you are presenting part of an unpublished archive, then constructing most of your .bib entry is fairly straightforward. "Letter-like" misc entries follow many of the same con-

ventions as letter entries presenting published material. Titles are of the form `Author to Recipient`, and further information can be given in the titleaddon field, including the abbreviation "n.d." (or `\bibstring{nodate}`) for undated examples. The place where a letter was written can go in origlocation, while the note, organization, institution, and location fields (in ascending order of generality) allow the specification of which manuscript collection now holds the letter, though the *Manual* specifies (14.227) that well-known depositories don't usually need a city, state or country specified. (The traditional misc fields are all still available, also.) Both the long and short note forms can use the same title, but in both cases you may need to use the `\headlesscite` command to avoid the awkward repetition of the author's name, though that name will always appear in the bibliography (creel:house). If the misc entry isn't a letter, remember that, as in article and review entries, words like interview or memorandum needn't be capitalized unless they follow a period — the automatic capitalization routines (with the title field starting with a lowercase letter [see dinkel:agassiz, spock:interview, and `\autocap`]) will ensure correctness. Also, please note that you can give additional information about the author in such entries by using the nameaddon field, providing your own square brackets if you're indicating that the author is pseudonymous, or parentheses if it's another sort of information of interest to your readers. The package options nameaddonformat and nameaddonsep can help here, as well. See sections 4.4.3 and 4.4.2, below.

Now for the subtleties. First, the *Manual* (14.224) allows in these entries, as it does in documentation generally "if numerous dates occur" (9.35), for a more streamlined presentation of dates using the day-month-year form, different from the standard American month-day-year. In letter entries you use the origdate field to give the date of individual letters, and it is always presented in the more streamlined form. Here, the same field will do exactly the same thing, though with the added wrinkle that if you'd prefer to use the standard day-month-year form you can, simply by putting the date into the date field instead. (Please choose one only in misc entries with an entrysubtype — in letter entries the date refers to the published collection.) Again just as in letter entries, if you want to include the day-month-year in a short note, I have provided the `\letterdatelong` command for inclusion in the postnote field of the citation command. (The standard biblatex command `\printdate` will work if you prefer the standard date form.)

Second, some material (roosevelt:speech) includes a venue for the event recorded in the archive, so I have added the venue field, which prints *before* the date, with the origlocation appearing after it. Somewhat confusingly, in published letters the origlocation itself prints before the date, rather than after, so if the inconsistency between published and unpublished letters bothers you then you could conceivably use venue instead of origlocation for that purpose here.

Finally, a few further notes. First, please be aware that defining an entrysubtype activates the automatic capitalization mechanism in the title field of misc entries, on which see `\autocap` in section 4.3.1 below. Second, and again as with letter entries, the *Manual* (14.222) suggests that bibliography entries contain only the name of the manuscript collection, unless only one item from that collection is cited. The crossref field can be used, as well as the skipbib option, for preventing the individual items from turning up in the bibliography. Obviously, this is a matter for your discretion, and if you're using only short notes (see the short option, section 4.4.3 below), you may feel the need to include more information in the note if the bibliography doesn't contain a full reference to an individual item. Third, the *Manual* offers several examples of specific location information for pieces from an archive, some of which appears *before* the main archive name, and some of which appears *after* it. I assume this may depend on the exact nature of the archive itself, but in any case you can try the type or howpublished fields for the first case and the number field for the second. Last, in all this class of archived material, the *Manual* (14.221) quite specifically requires more consistency within your own work than conformity to some external standard, so it is the former which you should pursue. I hope that biblatex-chicago-notes proves helpful in this regard. (See 14.211, 14.219, 14.221-231, 14.256, 14.264; creel:house, dinkel:agassiz, roosevelt:speech, shapey:partita, spock:interview.)

music This is one of three audiovisual entry types, and is intended primarily to aid in the presentation of musical recordings that do not have a video component, though it can also

include audio books (auden:reading). A DVD or VHS of an opera or other performance, by contrast, should use the **video** type instead, while an online music video will probably need an **online** entry. (Cf. online and video; handel:messiah, horowitz:youtube.) Because biblatex — and BibTeX before it — were designed primarily for citing book-like objects, some choices needed to be made in assigning the various roles found on the back of a CD to the fields in a typical .bib entry. I have also implemented several bibstrings to help in identifying these roles within entries. The 17th edition of the *Manual* once again revised its recommendations for this type, but fortunately the changes are additive, i.e., you can re-use 16th-edition citations but are encouraged to peruse the following guidelines to see if there's any information you might think of adding to bring your citations more into line with the spec.

These guidelines, in summary form, are:

author: composer, songwriter, or performer(s), they will be closely associated with the title, either before it at the head of the entry or, at your discretion, just after it (holiday:fool).

bookauthor: Somewhat like an author, but it will hold the name associated with a whole album rather than an individual piece, should both be present, and will therefore appear in close association with the booktitle, rather than the title (rihanna:umbrella).

editor, editora, editorb: conductor, director or performer(s). These will ordinarily follow the title of the work, though the usual useauthor and useeditor options can alter the presentation within an entry. Because these are non-standard roles, you will need to identify them using the following:

editortype, editoratype, editorbtype: The most common roles, all associated with specific bibstrings (or their absence), will be conductor, director, producer, and, oddly, none. The last is particularly useful when identifying the group performing a piece, as it usually doesn't need further specifying and this role prevents biblatex from falling back on the default editor bibstring. The 17th edition (14.263) also seems to favor, in some circumstances, using strings to identify individual performers, e.g., "vocalist" or "pianist," so even though there's no \bibstring associated with these types you can now provide them, or anything else you need, in whatever form ("vocalist" or "sung by") suits your citation.

note: This field can also hold contributors, perhaps collaborators or featured artists (holiday:fool, rihanna:umbrella).

title, booktitle, maintitle: As with the other audiovisual types, music serves as an analogue both to books and to collections, so the title will either be, e.g., the album title or a song title, in which latter case the album title would go into booktitle. If you wish to cite a song that, as may happen, isn't part of any larger collection, your entry will in such a case have only a title, which biblatex-chicago would normally interpret as an album title. You can now define an entrysubtype to let it know that the lone title is in fact a song (cf. naraya). The maintitle might be necessary for something like a box set of *Complete Symphonies*.

chapter: The 17th edition seems more keen on having track numbers for individual pieces, whether on a traditional format or on a streaming service. The chapter field is the place for this information, and biblatex-chicago will automatically prepend the localized string track (cf. holiday:fool, rihanna:umbrella).

publisher, series, number: These three closely-associated fields are intended for presenting the catalog information provided by the music publisher. The 17th edition generally only requires the series and number fields (nytrumpet:art), which hold the record label and catalog number, respectively. Alternatively, publisher would function as a synonym for series (holiday:fool), but there may be cases when you need or want to specify a publisher in addition to a label, as perhaps when a single

publisher oversees more than one label. You can certainly put all of this information into one of the above fields, but separating it may help make the .bib entry more readable.

pubstate: The pubstate field in music entries mainly has the usual meaning it has in other entry types, for which see the documentation of the field in section 4.2, below. If the field contains reprint, however, this has a special meaning in music entries, where it will transform the origdate from a recording date for an entire album into an original release date for that album, notice of which will be printed towards the end of a note or bibliography entry. No reprint \bibstring will be printed, as only the syntax of the reference will have been altered.

date, eventdate, origdate: The 17th edition of the *Manual*, like the 16th, considers music citations without a date to be "generally unacceptable" (14.263). Finding a date may take some research, but they will basically fall into two types, i.e., the date(s) of the recording or the copyright / publishing date(s). Recording dates go either in origdate (for complete albums) or eventdate (for individual tracks). The copyright or publishing dates go either in the date field (which applies to the current medium you are citing) or in the origdate field (which refers to the original release date). You may have noticed that the origdate has two slightly different uses — you can tell biblatex-chicago which sort you intend by using the string reprint in the pubstate field, which transforms the origdate from a recording date into an original release date. The style will automatically prepend the bibstring recorded to the eventdate or, in the absence of this pubstate mechanism, to the origdate, or even to both, but you can modify what is printed there using the userd field, which acts as a sort of date type modifier. In music entries, userd will be prepended to an eventdate if there is one, barring that to the origdate, barring that to a urldate, and absent those three to the date. (See floyd:atom, holiday:fool, nytrumpet:art.)

type, howpublished: As in all the audiovisual entry types, the type field holds the medium of the recording, e.g., vinyl, 33 rpm, 8-track tape, cassette, compact disc, mp3, ogg vorbis. The howpublished field, newly included for the 17th edition, can hold similar information "for streaming audio formats and downloads" (14.263). It can also, alternatively, hold the name of the streaming service, e.g., Spotify (cf. rihanna:umbrella).

The entries in notes-test.bib should at least give you a good idea of how this all works, and that file also contains an example of an audio book presented in a music entry. If you browse the examples in the *Manual* you will see the sheer variety of possibilities for presenting these sources, my intention being that judicious manipulation of .bib entries should allow you to make biblatex-chicago do what you want. Please let me know if I've ignored something you need. (Cf. 14.263–64; eventdate, origdate, userd; auden:reading, beethoven:sonata29, bernstein:shostakovich, floyd:atom, holiday:fool, nytrumpet:art, rubinstein:chopin.)

mvbook
mvcollection
mvproceedings
mvreference

The 17th edition of the *Manual* has deployed, in at least two contexts, a notable syntactic change in the presentation of works that form part of other, larger works. Generally, the order of presentation, in biblatex terms, has always been title – booktitle – maintitle, in increasing order of generality. In the vast majority of cases this order still holds, but in TV episodes, for one example, the recommendation now is to present the name of the series (booktitle) *before* the name of the episode (title). The **video** type (14.265) provides this by using an entrysubtype, tvepisode, which reverses the order for you in both long and short notes, and in the bibliography. The other context in which this reversal occurs is multi-volume works (14.116–22). Here, the preferred format, at least for notes, appears to be maintitle – [book]title or, when all three titles are present, title – maintitle – booktitle. The *Manual* doesn't carry this reordering through with absolute consistency, but I think it important at least to offer it as a possibility to users of biblatex-chicago. Reluctant as I am simply to change the data model and provide non-standard entry types, the least invasive method seemed, and seems, to me to be to modify the **mv*** entry types while maintaining backward compatibility with users' current deployments of these types.

So, while these types are no longer aliased to the entry type that results from removing the "mv" from their names, any .bib entries using them should, without modification, continue to function as they always have. Should you wish to ignore the new syntax of presentation, and there are still examples in the *Manual* which do just that, then no changes are necessary. These entries will still function, assuming you are using Biber, as the target of cross-references from other entries, the title of the **mv*** entry *always* providing a maintitle for the entry referencing it. (If you want to provide a booktitle for the referencing entry, please use another entry type, e.g., **collection** for **incollection** or **book** for **inbook**. These distinctions are particularly important to the correct functioning of the abbreviated references that biblatex-chicago, in various circumstances, provides. Please see the documentation of the **crossref** field in section 4.2, below.)

Also unchanged is the requirement, when multi-volume works are presented in the reference apparatus, that any dates should be appropriate to the specific nature of the citation. This means that a date range that is right for the presentation of a multi-volume work in its entirety isn't right for citing, e.g., a single volume of that work which appeared in one of the years contained in the date range. Because child entries will by default inherit all the date fields from their parent (including the endyear of a date range), I have turned off the inheritance of date and origdate fields from all of the **mv*** entry types to any other entry type. When the dates of the parent and of the child in such a situation are exactly the same, then this unfortunately requires an extra field in the child's .bib entry. When they're not the same, as will, I believe, often be the case, this arrangement saves a lot of annoying work in the child entry to suppress wrongly-inherited fields. Other sorts of parent entries aren't affected by this, and of course you must be using Biber for the settings to apply.

maintitle   Should you wish to employ the new, maintitle-first syntax, then you'll need to use the maintitle relatedtype. In its simplest usage, to document one volume of a multi-volume set, you would have, e.g., an mvcollection entry with relatedtype maintitle, and a related field pointing to a collection entry. When you cite the mvcollection entry itself, you'll get a long note like *MVCollTitle*, vol. 1, *CollTitle*, and a short note like *MVCollTitle*, vol. 1., or, with a postnote field, *MVCollTitle*, 1:12, as the specification requires. If you wanted to cite one essay in the collection, then you would, additionally, need an incollection entry with the maintitle relatedtype and a related field pointing to the mvcollection entry already mentioned, so you're creating a chain of three different related entries but presenting them in one reference. It's important to keep in mind here that, in effect, you're *not* actually citing the mvcollection entry, but the one volume of it represented by the collection entry, or indeed an essay in that one volume. Please consult the *Manual* (14.116–22), and also see harley:ancient:cart, harley:cartography, and harley:hoc for the "old style" presentation with abbreviated cross references using the crossref field, harleymt:ancient:cart, harleymt:cartography and harleymt:hoc for the new presentation using the maintitle relatedtype, and also plato:timaeus:gr for an example of a three-work maintitle chain starting with a bookinbook entry.

The documentation of the maintitle relatedtype in section 4.2.1 contains all the details, but there are several things I should like to mention here. First, while you can happily mix these two methods of presentation in your documents, please don't mix them within individual entries, which means that if you are using a crossref field to an mvcollection entry in a collection entry, say, and the collection entry is itself the target of the mvcollection entry's related field, please be careful not to cite that collection entry independently, as it can lead to unexpected results. (If things don't look right to you, try eliminating the use of crossref entirely from these related chains and see if that helps, then send me a bug report if it does.) This restriction also means that, although the *Manual* prefers the maintitle-first format in notes and allows either syntax in the bibliography, nonetheless with biblatex-chicago whichever syntax you choose for the notes will also appear in the bibliography. Second, if you want to use a three-work chain to cite one part of one volume, then this is possible only by using the following entry types: bookinbook, inbook, incollection, inproceedings, and letter. All two-work chains must start with one of the mv* types. Third, as might be apparent from the previous list, mvreference entries are special, in that their related field should point to an inreference entry if you want to cite an entry in an "alphabetically arranged work", or to a reference entry otherwise.

Fourth, please remember that, as these are citations not of an mv* entry but rather of that entry's related field, any volumes field in the former won't be printed by default. You can change this by setting the hidevolumes option to false either in the preamble or in the options field of the entry referenced by the mv* entry's related field. Finally, if you look at the "mt" variants of the harley* entries mentioned above, you'll see that harleymt:hoc has both subsidiary volumes included in its related field. You can create a separate mv* entry pointing to each of it's subsidiary volumes, or you can list all of those volumes in one mv* entry's related field and biblatex-chicago will create separate clones for each volume listed, clones with a standardized entry key looking like "mventrykey-singlevolumeentrykey," which you should then use for your citations. The original mventrykey, in this case, refers merely to the original mv* entry, as though it had never had a related field, though you do need to cite (or \nocite) it somewhere in your document to make the single-volume clones available in your reference apparatus. The mechanism's designed to save you some typing in common scenarios; please see all of the (multifarious) details in section 4.2.1, below.

online   One of the features of the 17th edition of the *Manual* is the considerably extended, but still scattered, treatment of online materials (8.189–92, 14.6–18, 14.159–63, 14.175–76, 14.187, 14.189, 14.205–10, 14.233). The principles of that treatment have changed somewhat, as the *Manual* now places greater emphasis on the *location* of a source, which can in many cases outweigh, as far as choosing an entry type goes, the *nature* of the source. Working out the correspondences between online sources and biblatex-chicago entry types can, therefore, be tricky, so I have included table 1 summarizing the increasingly detailed instructions in the *Manual*, along with some further annotations here that might help to clarify it.

The basic principle, as I've cited in the penultimate entry of table 1, is that "the title of a website that is analogous to a traditionally printed work but does not have (and never had) a printed counterpart can be treated like the titles of other websites, subject to editorial discretion" (14.206). This means that an intrinsically online entry like stenger:privacy (citing CNN.com) need no longer be an article but can be presented in an online entry. (The same principle applies to wikiped:bibtex, but because of the code facilitating presentation of alphabetized entries in reference works, it's best in this case to keep the inreference entry but add an entrysubtype so that the title is presented as it would be in an online entry.) The corollary of the principle, as the first entry in table 1 suggests, is that an online edition of a printed work will generally require the same entry type as that printed work itself would. Blogs are, therefore, somewhat anomalous in requiring the various periodical types, though the *Manual* does specify that if you're not sure whether a website is a blog, then it probably requires the online type (14.206). Social media, on the other hand, are very much subject to the first principle, requiring online entries no matter whether the citation is of text, a photo, or a video. Without pretending that all of the correspondences flow deductively from the basic principles, I hope that the table might simplify most of your choices. If something remains unclear, please let me know and I'll see if I can improve it.

A few more notes are in order. I designed the relatedtype commenton to facilitate citation of online comments, though it works slightly differently in the two entry types in which it is available, online and review. In both types it allows you to mimic thread structure by citing a chain of replies to comments on posts, etc., all in a single entry, while also simplifying your .bib entries. This simplification works differently depending on whether the comment itself has no specific title, as always in review entries, or does have such a title, as especially in online social media entries. In the former case, the related apparatus allows you not to provide a title at all, but in the latter you still need a title, which will be followed by the relatedstring. In these latter entries, the *only way to cite such comments* is by using the commenton relatedtype (licis:diazcomment). If, in online entries, you decided *not* to use commenton in an entry like braun:reply, and simply use a specially-crafted titleaddon field, you lose the possibility of having two dates in the entry, one for the comment and one for the original post, though to be fair it does end up looking like the example in 14.210, where it is ambiguous to which part of the citation the date applies.

As for the thread structure, I've not tested how far down the rabbit hole you can go, but a series of entries linked one to the next by this relatedtype will all turn up if you cite the

Table 1: Online materials and notes & bibliography entry types

Online Material	Entry Type	CMS Ref.	Sample Entry	Notes
Online edition of trad. publ. matter.				Use the same entry type as you would choose were you citing it from a printed source.
	@Book	14.161-62	james:ambassadors	CMS prefers (scanned) original page numbers to reflowable text.
	@Article @Review	14.175	black:infectious	If no "suitable URL" is available, e.g., if it points to a generic portal page rather than to an abstract, use the name of the commercial database in an addendum field instead.
Blogs		14.208		
Entire	@Periodical		amlen:wordplay	The maintitle field holds the larger publication of which the blog is a part.
Single post	@Article		amlen:hoot	
Comment	@Review		viv:amlen	You can also use the commenton relatedtype for this.
Social Media	@Online			This includes anything — posts, photos, videos — on these and similar sites. In other words, the *location* of the material defines its treatment.
Mailing list or forum post		14.210	powell:email	Posts on private lists are to be treated as "personal communications," using @Misc w/ entrysubtype.
Facebook		14.209	diaz:surprise	
Twitter			obrien:recycle	
Instagram			souza:obama	
Comments / replies		14.210	braun:reply	The commenton relatedtype is *required* for this, and for the next entry, too.
		14.209	licis:diazcomment	
Online Multimedia				
Online video	@Online	14.267	pollan:plant	This category includes TED talks and most informal videos on YouTube and similar sites.
Online video, from a trad. journal	@Article		kessler:nyt	You can use @Online, but this requires special formatting in the note or titleaddon field.
Published films in an archive	@Video		weed:flatiron	
Podcasts	@Audio		danforth:podcast	Note the eventdate of the individual episode.
Archival audio	@Misc w/ entrysubtype	14.264	roosevelt:speech	Can have both a venue and an origlocation.
Streaming Media				
TV / Film	@Video	14.265	mayberry:brady	The streaming service is supplied by the URL. The tvepisode entrysubtype is new in the 17th edition.
Music	@Music	14.263	rihanna:umbrella	The streaming service is supplied by the howpublished field.
News / Interviews	@Article @Review	14.213	bundy:macneil	Network information goes in the usera field.
Websites	@Online	14.206-7	evanston:library stenger:privacy	An online source "analogous to a traditionally printed work but [which] does not have (and never had) a printed counterpart" may now use an @Online entry, at your discretion.
Reference works, cited by alphabetized entry	@InReference w/ entrysubtype	14.233	wikiped:bibtex	As above, you can choose the @Online treatment of the title, but it's best achieved using an @InReference entry w/ entrysubtype.
Scientific databases	@Dataset	14.257	genbank:db	New in this release.

first in the chain, though of course you can use the technique merely as a convenient way to structure and simplify your .bib file, without creating chains longer than 2 entries. The default connecting string is the localized commenton, but you can use relatedstring to change it to "reply to" or whatever else you need. Please see the documentation of this relatedtype in section 4.2.1, and also diaz:surprise and licis:diazcomment.

In general, constructing an online .bib file entry is much the same as in biblatex. The title field would contain the title of the page, the organization field could hold the title or owner of the whole site. If there is no specific title for a page, but only a generic one, then such a title should go in titleaddon, not forgetting to begin that field with a lowercase letter so that capitalization will work out correctly. It is worth remarking here, too, that the *Manual* (14.12–13) prefers, if they're available, revision dates to access dates when documenting online material. Indeed, given how rapidly online sources may change (14.191, 14.209, 14.233), a time stamp may often be necessary further to specify a revision date (urldate) or the date of a comment or reply (date). This time specification should be added to the date field using biblatex's standard format, i.e., 2008-07-01T10:18:00. If a time zone is needed, then a separate timezone or urltimezone field is the best way, as it allows you to provide the initialisms that the *Manual* prefers (10.41, 14.191). See date, timezone, urldate, and userd in section 4.2, below.

patent The *Manual* is very brief on this subject (14.258), but very clear about which information it wants you to present, so such entries may not work well with other biblatex styles. The important date, as far as Chicago is concerned, is the filing date. If a patent has been filed but not yet granted, then you can place the filing date in either the date field or the origdate field, and biblatex-chicago-notes will automatically prepend the bibstring patent-filed to it. If the patent has been granted, then you put the filing date in the origdate field, and you put the date it was issued in the date field, to which the bibstring patentissued will automatically be prepended. You can place additional information in the addendum field if desired, and it will be printed in close association with the dates. The patent number goes in the number field, and you should use the standard biblatex bibstrings in the type field. Though it isn't mentioned by the *Manual*, biblatex-chicago-notes will print the holder after the author, if you provide one. Finally, the style automatically capitalizes patent titles sentence-style, so if you need to keep a word capitalized then you should wrap it in curly braces. See petroff:impurity.

performance The 17th edition of the *Manual* includes a new section (14.266) on citing live performances, and even though such references can usually be limited to the main text it may sometimes be useful to include them in notes. Since biblatex provides the **performance** type, albeit without using it in its standard styles, I though it might be useful to define it for biblatex-chicago, particularly as the other option for such material is the misc entry without any entrysubtype, and that entry type is already somewhat overloaded, though you can still use it if you wish.

Such entries will generally have a title, a venue, a location for the venue, and a date for the performance, along with a possible plethora of authorial and/or editorial roles depending on which sorts of contributor(s) you wish to emphasize in the citation. The editor[abc] and editor[abc]type fields should be most helpful here. I have included strings for choreographer in all localization files, but for others you may need to provide them in the editor[abc]type fields as you wish them printed — biblatex-chicago will automatically capitalize any that start with a lowercase letter.

periodical This is the standard biblatex entry type for presenting an entire issue of a periodical, rather than one article within it. It has the same function in biblatex-chicago-notes, and in the main uses the same fields, though in keeping with the system established in the article entry type (which see) you'll need to provide entrysubtype magazine if the periodical you are citing is a "newspaper" or "magazine" instead of a "journal." Also, remember that the note field is the place for identifying strings like "special issue," with its initial lowercase letter to activate the automatic capitalization routines. (See *Manual* 14.178; good:wholeissue.)

It is worth noting a few things. First, the titleaddon field is now available in these entries, but as the title here is analogous to the journaltitle in article or review entries, the new jtitleaddon option (section 4.4.2) governs the punctuation separating the titleaddon from the title. Second, the special biblatex field shortjournal allows you to present shortened journaltitles in article, review, and periodical entries, as well as facilitating the creation of lists of journal abbreviations in the manner of a shorthand list. Because the periodical type uses the title field instead of journaltitle, biblatex-chicago automatically copies any shorttitle field, if one is present, into shortjournal. Please see the documentation of **shortjournal** in section 4.2 for all the details on how this works. Finally, the periodical type is the place for citing whole blogs, rather than individual blog posts, which require either an article or a review entry. In such citations the 17th edition (14.208) recommends that you include the name of any larger (usually periodical) publication of which the blog is a part. The maintitle field (with mainsubtitle and maintitleaddon, if needed) is the place for it. Cf. amlen:wordplay.

proceedings This is the standard biblatex and BIBTEX entry type, now also including the eventdate, eventtitle, eventtitleaddon, and venue fields for identifying the event that produced the proceedings. The package can automatically provide abbreviated references in notes and bibliography when you use a crossref or an xref field. The functionality is not enabled by default, but you can enable it in the preamble or in the options field using the booklongxref option. Please see **crossref** in section 4.2 and booklongxref in section 4.4.2, below.

reference This entry type is aliased to collection by the standard biblatex styles, but I intend it to be used in cases where you need to cite a reference work but not an alphabetized entry or entries in that work. This could be because it doesn't contain such entries, or perhaps because you intend the citation to appear in a bibliography rather than in notes. Indeed, the only differences between it and inreference are the lack of a lista field to present an alphabetized entry, and the fact that any postnote field will be printed verbatim, rather than formatted as an alphabetized entry. (See mla:style for an example of a reference work that uses numbered sections rather than alphabetized entries, and that appears in the bibliography as well.)

report This entry type is a biblatex generalization of the traditional BIBTEX type techreport. Instructions for such entries are rather thin on the ground in the *Manual* (8.186, 14.220), so I have followed the generic advice about formatting it like a book, and hope that the results conform to the specification. At least one user has indicated a need, now filled, for an unpublished entrysubtype, which prints the title inside quotation marks instead of in italics, but affects nothing else. This detail aside, the type's main peculiarities are the institution field in place of a publisher, the type field for identifying the kind of report in question, the number field closely associated with the type, and the isrn field containing the International Standard Technical Report Number of a technical report. As in standard biblatex, if you use a techreport entry, then the type field automatically defaults to \bibstring{techreport}. As with booklet and manual, you can also use a book entry, putting the report type in note and the institution in publisher. (See herwign:office.)

review As its name suggests, the review entry type was designed for reviews published in periodicals, and if you've already read the article instructions above — if you haven't, I recommend doing so now — you'll know that review serves as well for citing other sorts of material with generic titles, like letters to the editor, obituaries, interviews, online comments and the like. The primary rule is that any piece that has only a generic title, like "review of …," "interview with …," or "obituary of …," calls for the review type. Any piece that also has a specific title, e.g., "'Lost in BIBTEX,' an interview with …," requires an article entry. (This assumes the text is found in a periodical of some sort. Were it found in a book, then the incollection type would serve your needs, and you could use title and titleaddon there. While we're on the topic of exceptions, the *Manual* includes an example (14.213) where the "Interview" part of the title is considered a subtitle rather than a titleaddon, said part therefore being included inside the quotation marks and capitalized accordingly. Not having the journal in front of me I'm not sure what prompted that decision, but biblatex-chicago would obviously have no difficulty coping with such a situation.)

Once you've decided to use review, then you need to determine which sort of periodical you are citing, the rules for which are the same as for an article entry. If it is a "magazine" or a "newspaper", then you need an entrysubtype magazine, or the synonymous entrysubtype newspaper. The generic title goes in title and the other fields work just as as they do in an article entry with the same entrysubtype, including the substitution of the journaltitle for the author if the latter is missing. (See 14.190–91, 14.195–96, 14.201–4, 14.213; barcott:review, bundy:macneil, Clemens:letter, gourmet:052006, kozinn:review, nyt:obittrevor, nyt:trevorobit, unsigned:ranke, wallraff:word.) If, on the other hand, the piece comes from a "journal," then you don't need an entrysubtype. The generic title goes in title, and the remaining fields work just as they do in a plain article entry. (See 14.202; ratliff:review.)

Biblatex-chicago also, at the behest of Bertold Schweitzer, supports the relatedtype reviewof, which allows you to use the related mechanism to provide information about the work being reviewed, thereby simplifying how much information you need to provide in the reviewing entry. In particular, it relieves you of the need to construct title or titleaddon fields like: review of \mkbibemph{Book Title} by Author, as the related entry's title automatically provides the title in the review type and the titleaddon in the article type, with the related mechanism providing the connecting string. This may be particularly helpful if you need to cite multiple reviews of the same work; please see section 4.2.1 for further information.

Most of the onerous details are the same as I described them in the **article** section above, but I'll repeat some of them briefly here. If anything in the title needs formatting, you need to provide those instructions yourself, as the default is completely plain. (As just mentioned, the related mechanism provides this automatically.) Author-less reviews are treated just like similar newspaper articles — in short notes and in the bibliography the journaltitle replaces the author and heads the entry, while in long notes the title comes first. The sorting of such entries is an issue, solved if you use Biber as your backend, and otherwise requiring manual intervention with, e.g., a sortkey (14.204; gourmet:052006, nyt:trevorobit, unsigned:ranke, and see \DeclareSortingTemplate in section 4.4.1, below.). As in misc entries with an entrysubtype, words like "interview," "review," and "letter" only need capitalization after a full stop, i.e., ordinarily in a bibliography and not a note, so biblatex-chicago-notes automatically deals with this problem itself if you start the title field with a lowercase letter. The file notes-test.bib and the documentation of \autocap will provide guidance here.

One detail of the review type is fairly new, and in particular has changed between the 16th and 17th editions of the *Manual*. As I mentioned above, blogs are best treated as articles with magazine entrysubtype, whereas comments on those blogs — or replies to those comments, etc. — need the review type with the same entrysubtype. (Neither need appear in the bibliography.) What they also need is a date closely associated with the comment (14.208; ac:comment), so I have included the eventdate in review entries for just this purpose. It will be printed just after the author and before the title. If you need a time stamp in addition, as may frequently be the case with multiple contributions by the same author to a single thread, then you should now use the standard biblatex time-stamp format (e.g., 2008-07-01T10:18:00) in the eventdate field itself, which biblatex-chicago will format and print appropriately. Please see the documentation concerning time stamps in section 4.2, s.v. date. This change allows the nameaddon field to revert to its primary use, which is to provide extra information about the author. In blog comments, this could include the commenter's geographical location, which you need to enclose in parentheses, as I've removed the automatic square brackets from this field to allow it this more general usefulness. You can, of course, still provide your own square brackets in review entries to indicate pseudonymous authorship, which is the standard function of nameaddon in most entry types. The package options nameaddonformat and nameaddonsep can help here, as well. See sections 4.4.3 and 4.4.2, below.

In this context I should mention a small change to the default behavior of review entries when they utilize a crossref or xref field, as is really only useful when the entry is a blog comment, as otherwise there won't generally be any fields worth inheriting from the reviewed entry. Assuming the default values of the biblatex-chicago option longcrossref,

blogurl — the driver now explicitly tests if the reviewed entry has already been cited, and accordingly shortens the reviewing citation, as the *Manual* (14.208) suggests (viv:amlen). (This would be incorrect for, say, a book review, so you should either not use a crossref or xref field there, or change the state of the longcrossref option — cf. the documentation starting on page 31, below.) You can, if you wish, and while we're on this subject, set the preamble option blogurl to allow your child comments to inherit the URL from the parent blog.

Another recent addition is the relatedtype commenton, which allows you to simplify your .bib entries in much the same way as the reviewof relatedtype does, i.e., it constructs your title field for you (which the crossref mechanism doesn't do). It further allows you to mimic thread structure by citing a chain of replies to comments on blogs, etc., all in a single entry, while also simplifying your .bib entries. I've not tested how far down the rabbit hole you can go, but a series of entries linked one to the next by this relatedtype will all turn up if you cite the first in the chain, though of course you can use the technique merely as a convenient way to structure and simplify your .bib file, without creating chains longer than 2 entries. The default connecting string is the localized commenton, but you can use relatedstring to change it to "reply to" or whatever else you need. Please see the documentation of this relatedtype in section 4.2.1, and also ellis:blog and ac:comment. Note also that this way of structuring your .bib file is by no means required in review entries, though if you want to cite replies and comments to social media threads, where you need the online entry type, you will need to use this relatedtype.

Two more notes. For the reasons I explained in the article docs above, I have brought the article and review entry types into line with most of the other types in allowing the use of the namea and nameb fields in order to associate an editor or a translator specifically with the title. The editor and translator fields, in strict homology with other entry types, are associated with the issuetitle if one is present, and with the title otherwise. The usual string concatenation rules still apply — cf. editor and editortype in section 4.2, below.

Finally, the special biblatex field shortjournal allows you to present shortened journaltitles in review entries, as well as in article and periodical entries, and it facilitates the creation of lists of journal abbreviations in the manner of a shorthand list. Please see the documentation of **shortjournal** in section 4.2 for all the details on how this works.

standard — In older releases it was fairly straightforward to present published national or international standards using a book entry, but with some additional specifications now included in the 17th edition of the *Manual* (14.259) I think it might be helpful to provide a separate entry type. The **standard** type has long existed in biblatex, though none of its included styles use it. In biblatex-chicago constructing such an entry is mostly straightforward. The organization responsible for the standard goes in organization, the title in title, and the series and number fields provide the ID of the standard. The date field generally provides the publication date, though for some standards there may also be a later reaffirmation date (or similar), for which you can use the eventdate.

Now, for the peculiarities. In the bibliography, the organization will appear at the head of the entry, and will be reprinted as the publisher. If you wish to provide a shortened version for the second appearance, then the publisher field is the place for it. In long notes, the entry starts with the title, so there the code prefers the organization as publisher, because its shortened version may not be immediately recognizable. In short notes, only the title will appear (along with any pre or postnote fields, obviously). You can use the author field in addition to the organization, but this is unnecessary. If you absolutely must have the organization or author appear at the head of long and short notes, then providing any entrysubtype whatsoever will accomplish this. Any named editor or namec will, as per the specification, *not* appear at the head of entries. You can really only alter this by using a book entry, instead. (Cf. w3c:xml.)

Finally, it is distinctly possible that an entry with two dates will need somehow to specify just what sort of dates are involved. The usual biblatex-chicago method is the userd field, and here that field will act as a date-type for the date field itself. For the eventdate, you'll need to use howpublished, which I have commandeered for this purpose in a few other entry types, as well. (Cf. niso:bibref and **howpublished** in section 4.2, below.)

suppbook   This is the entry type to use if the main focus of a reference is supplemental material in a book or in a collection, e.g., an introduction, afterword, or forward, either by the same or a different author. In previous releases of biblatex-chicago these three just-mentioned types of material, and only these three types, could be referenced using the introduction, afterword, or foreword fields, a system that required you simply to define one of them in any way and leave the others undefined. The macros don't use the text provided by such an entry, they merely check to see if one of them is defined, in order to decide which sort of pre- or post-matter is at stake, and to print the appropriate string before the title in long notes, short notes, list of shorthands, and bibliography. I have retained this mechanism both for backward compatibility and because it works without modification across multiple languages, but have also added functionality which allows you to cite any sort of supplemental material whatever, using the type field. Under this system, simply put the nature of the material, including the relevant preposition, in that field, beginning with a lowercase letter so biblatex can decide whether it needs capitalization depending on the context. Examples might be "preface to" or "colophon of." (Please note, however, that unless you use a \bibstring command in the type field, the resultant entry will not be portable across languages.)

There are a few other rules for constructing your .bib entry. The author field refers to the author of the introduction or afterword, while bookauthor refers to the author of the main text of the work, if the two differ. The *Manual* requires the inclusion of the page range of the part in question, though *only* in the bibliography. I have followed this advice literally, so the pages field of a suppbook entry won't automatically appear in a long note. If you wish to include those pages in a note, then you'll need to repeat them in the postnote field of the citation command.

Finally, if the focus of the reference is the main text of the book, but you want to mention the name of the writer of an introduction or afterword for bibliographical completeness, then the normal biblatex rules apply, and you can just put their name in the appropriate field of a book entry, that is, in the foreword, afterword, or introduction field. (See *Manual* 14.110; polakow:afterw, prose:intro).

suppcollection   This fulfills a function analogous to suppbook. Indeed, I believe the **suppbook** type can serve to present supplemental material in both types of work, so this entry type is an alias to suppbook, which see.

suppperiodical   This type is intended to allow reference to generically-titled works in periodicals, such as regular columns or letters to the editor. Biblatex-chicago-notes provides the review type for this purpose, and you can use either of these, as I've added suppperiodical as an alias of review. Please see above under **review** for the full instructions on how to construct a .bib entry for such a reference.

unpublished   The unpublished entry type works largely as it does in standard biblatex, though it's worth remembering that you should use a lowercase letter at the start of your note field (or perhaps an \autocap command in the somewhat contradictory howpublished, if you have one) for material that wouldn't ordinarily be capitalized except at the beginning of a sentence. Thanks to a bug report by Henry D. Hollithron, such entries will print information about any editor, translator, compiler, etc., that you include in the .bib file. Also, conforming to the indications of the *Manual*, and thanks to the prompting of Jan David Hauck, you can use the venue, eventdate, eventtitle, and eventtitleaddon fields further to specify unpublished conference papers and the like (14.216–18; nass:address).

video   This is the last of the three audiovisual entry types, and as its name suggests it is intended for citing visual media, be it films of any sort or TV shows, whether broadcast, on the Net, on VHS, DVD, or Blu-ray, though it will serve as well, I think, for radio broadcasts of plays or drama serials. As with the music type discussed above, certain choices had to be made when associating the production roles found, e.g., on a DVD, to those bookish ones provided by biblatex. The 17th edition of the *Manual* once again revised its recommendations for this type, but fortunately the changes are additive, i.e., you can re-use 16th-edition

citations but are encouraged to peruse the following guidelines to see if there's any information you might think of adding to bring your citations more into line with the spec. Here are the main guidelines:

author: This will not infrequently be left undefined, as the director of a film should be identified as such and therefore placed in the editor field with the appropriate editortype (see below). You will need it, however, to identify the composer of, e.g., an oratorio on VHS (handel:messiah), or perhaps the provider of commentaries or other extras on a film DVD (cleese:holygrail).

editor, editora, editorb = director or producer, or possibly the performer or conductor in recorded musical performances. These will ordinarily follow the title of the work, though the usual useauthor and useeditor options can alter the presentation within an entry. Because these are non-standard roles, you will need to identify them using the following:

editortype, editoratype, editorbtype: The most common roles, all associated with specific bibstrings (or their absence), will likely be director, producer, and, oddly, none. The last is particularly useful if you want to identify performers, as they usually don't need further specifying and this role prevents biblatex from falling back on the default editor bibstring. Any other roles you want to emphasize, even if there is no pre-defined \bibstring, can be provided here, and will be printed as-is, contextually capitalized. (Cf. hitchcock:nbynw.)

title, titleaddon, booktitle, booktitleaddon, maintitle: As with the other two audiovisual types, video serves as an analogue both to books and to collections, so the title may be of a whole film DVD or of a TV series, or it may identify one episode in a series or one scene in a film. In the latter cases, the title of the whole would go in booktitle. The booktitleaddon field is the place for specifying the season and/or episode number of a TV series, while the titleaddon is for any information that needs to come between the title and the booktitle (american:crime, cleese:holygrail, friends: leia, handel:messiah, hitchcock:nbynw, mayberry:brady). As in the music type, a maintitle may be necessary for a boxed set or something similar.

entrysubtype: If, for some reason, you want to cite an individual episode or scene without reference to any larger unit, then your entry will contain only a title, which biblatex-chicago would normally interpret as the title of a complete film or TV series. In such a case, you'll need to define an entrysubtype to let it know that the lone title is such a sub-unit. In quite a different syntactic transformation, the 17th edition (14.265) now recommends that, when presenting episodes from a TV series, the name of the series (booktitle) comes before the episode name (title). The exact string tvepisode in the entrysubtype field achieves this reversal, which includes using the booktitle as a sorttitle in the bibliography and also as the labeltitle in short notes.

usera: When citing recordings of TV shows, the 17th edition now wants you to include the TV network for the original broadcast, and the usera field is the place for it. Biblatex-chicago has long used this field for this same purpose in article, periodical, and review entries with a magazine entrysubtype, so its inclusion here can at least hope to benefit from that prior acquaintance. It will appear after the broadcast date, i.e., the eventdate, and will be separated from it by the \bibstring "on."

date, eventdate, origdate: As with music entries, in order to follow the specifications of the *Manual*, we need to provide three separate date fields for citing video sources, but their uses differ somewhat between the two types. In both, the date will generally provide the publishing or copyright date of the medium you are referencing. The eventdate will most commonly present either the broadcast date of a particular TV program, or the recording/performance date of, for example, an opera on DVD. The style will automatically prepend the bibstring broadcast to such a date, though you can use the userd field to change the string printed there. (Absent an eventdate, the userd field in video entries will modify the urldate, and absent those two it will modify the date.) The origdate has more or less the same function, and

appears in the same places, as it does in standard book-like entries, providing the date of first release of a film, though there isn't any reprint string associated with it in this entry type. Cf. friends:leia, handel:messiah, hitchcock:nbynw.

type: As in all the audiovisual entry types, the type field holds the medium of the title, e.g., 8 mm, VHS, DVD, Blu-ray, MPEG.

As with the music type, entries in notes-test.bib should at least give you a good idea of how all this works. (Cf. 14.265, 14.267; loc:city, weed:flatiron.)

4.2 Entry Fields

The following discussion presents, in alphabetical order, a complete list of the entry fields you will need to use biblatex-chicago-notes. As in section 4.1, I shall include references to the numbered paragraphs of the *Chicago Manual of Style*, and also to the entries in notes-test.bib. Many fields are most easily understood with reference to other, related fields. In such cases, cross references should allow you to find the information you need.

addendum As in standard biblatex, this field allows you to add miscellaneous information to the end of an entry, after publication data but before any url or doi field. In the patent entry type (which see), it will be printed in close association with the filing and issue dates. In all other entry types this information will come *after* any pages or postnote references present in long notes, allowing you in particular to use the field to identify a particular type of book-like publication when such data won't fit well in another part of an entry. In any entry type, if your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and the style will take care of the rest. Cf. note. (See *Manual* 14.114, 14.159–63; davenport:attention, natrecoff:camera.)

afterword In most circumstances, this field will function as it does in standard biblatex, i.e., you should include here the author(s) of an afterword to a given work. The *Manual* suggests that, as a general rule, the afterword would need to be of significant importance in its own right to require mentioning in the reference apparatus, but this is clearly a matter for the user's judgment. As in biblatex, if the name given here exactly matches that of an editor and/or a translator, then biblatex-chicago-notes will concatenate these fields in the formatted references.

As noted above, however, this field has a special meaning in the suppbook entry type, used to make an afterword, foreword, or introduction the main focus of a citation. If it's an afterword at issue, simply define afterword any way you please, leave foreword and introduction undefined, and biblatex-chicago-notes will do the rest. Cf. foreword and introduction. (See *Manual* 14.105, 14.110; polakow:afterw.)

annotation At the request of Emil Salim, biblatex-chicago-notes provides a package option (see annotation below, section 4.4.3) to allow you to produce annotated bibliographies. A recent feature request by Moritz Wemheuer referenced a StackExchange question which suggested that the possible uses for the annotation field could well be more extensive, appearing as it does at the very end of all entry types. I have therefore modified the annotation option so that you can print the field in the bibliography (=bib or =true, the default), in long notes (=notes), in both (=all), or in neither (=false). The two options bibannotesep and citeannotesep allow you to choose the separator between the rest of the entry and the annotation, and to choose a different one in notes and bibliography. The default formatting in the bibliography (vpar) is to print the annotation as a separate block using \par\nobreak\vskip\bibitemsep #1, while in long notes the default (period) is to print it simply as an additional field, separated by a period. The *Manual's* guidelines (14.64) allow for both these possibilities, and I have provided a range of others, for which you should consult the full documentation in section 4.4.2. (Note also that both options can be set globally or per-type in the preamble, or per-entry in the options field of individual entries. For specialized needs, of course, you can re-declare the format [\DeclareFieldFormat{annotation}] in your preamble, or redefine the \bibannotesep and \citeannotesep commands there.) In section 4.4.3 you will find instructions for

employing the `formatbib` and `entrybreak` options to give you fine-grained control over the formatting of the entire bibliography, particularly with regard to TeX's page-breaking algorithms. The aim is to remove, in most cases, any need for you to delve into the low-level commands involved in these algorithms.

annotator I have implemented this biblatex field pretty much as that package's standard styles do, even though the *Manual* doesn't actually mention it. It may be useful for some purposes. Cf. commentator.

author For the most part, I have implemented this field in a completely standard fashion. Remember that corporate or organizational authors need to have an extra set of curly braces around them (e.g., `{{Associated Press}}`) to prevent biber from treating one part of the name as a surname (14.84, 14.200; assocpress:gun, chicago:manual). If there is no author, then biblatex-chicago-notes will, in the bibliography and long notes, look in sequence, for a namea, an editor, a nameb, a translator, or a namec (i.e., a compiler) and use that name (or those names) instead, followed by the appropriate identifying string (esp. 14.103, also 14.76, 14.121, 14.126, 14.180; boxer:china, brown:bremer, harley:cartography, schellinger:novel, sechzer:women, silver:gawain, soltes:georgia). Biblatex's sorting algorithms will use the first of those names found, which should ensure correct alphabetization in the bibliography. (See \DeclareSortingTemplate in section 4.4.1, below.) In short notes, where the labelname is used, the order searched is somewhat augmented: shortauthor, author, shorteditor, namea, editor, nameb, translator, namec. (See \Declare-Labelname in section 4.4.1.)

In the rare cases when this substitution mechanism isn't appropriate, you have (at least) two options: either you can (chaucer:liferecords) put all the information into a note field rather than individual fields, or you can use the standard biblatex options useauthor= false, usenamea=false, useeditor=false, usenameb=false, usetranslator=false, and usenamec=false in the options field (chaucer:alt). If you look at the chaucer:alt entry in notes-test.bib, you'll notice that you only need to turn off the fields that are present in the entry, but please remember to use the new option usenamec instead of the old usecompiler, as the latter doesn't work as smoothly and completely as biblatex's own name toggles.

This system of options, then, can turn off biblatex-chicago-notes's mechanism for finding a name to place at the head of an entry, but it also very usefully adds the possibility of citing a work with an author by its editor, compiler or translator instead (14.104; eliot:pound), something that wasn't possible before. For full details of how this works, see the editortype documentation below. (Of course, in collection, periodical and proceedings entries, an author isn't expected, so there the chain of substitutions starts with namea and editor. Also, in article or review entries with entrysubtype magazine, the absence of an author triggers the use of the journaltitle in its stead. See those entry types for further details.)

NB: The *Manual* provides specific instructions for formatting the names of both anonymous and pseudonymous authors (14.79–82). In the former case, if no author is known or guessed at, then it may simply be omitted (virginia:plantation). The use of "Anonymous" as the name is "generally to be avoided," but may in some cases be useful "in a bibliography in which several anonymous works need to be grouped." If, on the other hand, "the authorship is known or guessed at but was omitted on the title page," then you need to use the authortype field to let biblatex-chicago-notes know this fact. If the author is known (horsley:prosodies), then put anon in the authortype field, if guessed at (cook:sotweed) put anon? there. (In both cases, biblatex-chicago-notes tests for these *exact* strings, so check your typing if it doesn't work.) This will have the effect of enclosing the name in square brackets, with or without the question mark indicating doubt. As long as you have the right string in the authortype field, biblatex-chicago-notes will also do the right thing automatically in the short note form.

In most entry types (except customc), the nameaddon field furnishes the means to cope with the case of pseudonymous authorship. If the author's real name isn't known, simply put pseud. (or \bibstring{pseudonym}) in that field (centinel:letters). If you wish

to give a pseudonymous author's real name, simply include it there, formatted as you wish it to appear, as the contents of this field won't be manipulated as a name by biblatex (lecarre:quest). If you have given the author's real name in the author field, then the pseudonym goes in nameaddon, in the form Firstname Lastname, pseud. (creasey: ashe:blast, creasey:morton:hide, creasey:york:death). This latter method will allow you to keep references to one author's work under different pseudonyms grouped together in the bibliography, as recommended by the *Manual*, though it is now recommended that, whichever system you employ, you include a cross-reference from one name to the other in the bibliography. You can do this using a customc entry (ashe:creasey, morton:creasey, york:creasey). Please see the entry on **nameaddon**, below, for circumstances where you may need to provide your own square brackets when presenting a pseudonym, and also the package options nameaddonformat and nameaddonsep in sections 4.4.3 and 4.4.2, below.

authortype In biblatex-chicago, this field serves a function very much in keeping with the spirit of standard biblatex, if not with its letter. Instead of allowing you to change the string used to identify an author, the field allows you to indicate when an author is anonymous, that is, when their name doesn't appear on the title page of the work you are citing. As I've just detailed under author, the *Manual* generally discourages the use of "Anonymous" as an author, preferring that you simply omit it. If, however, the name of the author is known or guessed at, then you're supposed to enclose that name within square brackets, which is exactly what biblatex-chicago does for you when you put either anon (author known) or anon? (author guessed at) in the authortype field. (Putting the square brackets in yourself doesn't work right, hence this mechanism.) The macros test for these *exact* strings, so check your typing if you don't see the brackets. Assuming the strings are correct, biblatex-chicago-notes will also automatically do the right thing in the short note form. Cf. author. (See 14.79–80; cook:sotweed, horsley:prosodies.)

The *Manual* doesn't clarify how to treat multiple works by the same author, in one or more of which their name doesn't appear on the title page. By default, biblatex-chicago will, after the first appearance in the bibliography, replace identical authors with the 3-em dash, regardless of any authortype field that may be present. If you want to distinguish between works certainly written by and works merely ascribed to a given author, then you can use the dashed option in the options field of individual entries, and possibly also a sortname, to get the results you want.

bookauthor For the most part, as in biblatex, a bookauthor is the author of a booktitle, so that, for example, if one chapter in a book has different authorship from the book as a whole, you can include that fact in a reference (will:cohere). Keep in mind, however, that the entry type for introductions, forewords and afterwords (suppbook) uses bookauthor as the author of title (polakow:afterw, prose:intro).

bookpagination This, a standard biblatex field, allows you automatically to prefix the appropriate string to information you provide in a pages field. If you leave it blank, the default is to print no identifying string (the equivalent of setting it to none), as this is the practice the *Manual* recommends for nearly all page numbers. Even if the numbers you cite aren't pages, but it is otherwise clear from the context what they represent, you can still leave this blank. If, however, you specifically need to identify what sort of unit the pages field represents, then you can either hand-format that field yourself, or use one of the provided bibstrings in the bookpagination field. These bibstrings currently are column, line, paragraph, page, section, and verse, all of which are used by biblatex's standard styles.

There are two points that may need explaining here. First, all the bibstrings I have just listed follow the Chicago specification, which may be confusing if they don't produce the strings you expect. Second, remember that bookpagination applies only to the pages field — if you need to format a citation's postnote field, then you must use pagination, which see (10.42–43, 14.147–56).

booksubtitle The subtitle for a booktitle. See the next entry for further information.

booktitle In the bookinbook, inbook, incollection, inproceedings, and letter entry types, the booktitle field holds the title of the larger volume in which the title itself is contained as one part.

It is important not to confuse this with the maintitle, which holds the more general title of multiple volumes, e.g., *Collected Works*. It is perfectly possible for one .bib file entry to contain all three sorts of title (euripides:orestes, plato:republic:gr). You may also find a booktitle in other sorts of entries (e.g., book or collection), but there it will almost invariably be providing information for the traditional BIBTEX cross-referencing apparatus, which I discuss below (**crossref**). This provision is now unnecessary, assuming you are using biber.

booktitleaddon   An annex to the booktitle. It will be printed in the main text font, without quotation marks. If your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago-notes will automatically do the right thing. The package and entry options ptitleaddon and ctitleaddon (section 4.4.2) allow you to customize the punctuation that appears before the booktitleaddon field.

chapter   This field holds the chapter number, mainly useful only in an inbook or an incollection entry where you wish to cite a specific chapter of a book (ashbrook:brain). It now also holds the track number of individual pieces of music, whether on a traditional format or on a streaming service (holiday:fool, rihanna:umbrella).

commentator   I have implemented this biblatex field pretty much as that package's standard styles do, even though the *Manual* doesn't actually mention it. It may be useful for some purposes. Cf. annotator.

crossref   This field is the standard BIBTEX cross-referencing mechanism, and biblatex has adopted it while also introducing a modified one of its own (xref). If you have used BIBTEX (or bibtex8) the crossref field works much the same as it always has, while xref attempts to remedy some of the deficiencies of the usual mechanism by ensuring that child entries will inherit no data at all from their parents. Section 2.4.1 of biblatex.pdf contains useful notes on managing cross-referenced entries, and section 3.15 explains some of the limitations of the traditional backends, which offer only a small subset of Biber's features. The functionality, discussed below, for abbreviating references in book, bookinbook, collection, and proceedings entries, and for using the mv* entry types to do so, will prove extremely difficult to replicate with the older backends, so if you plan on lots of cross-referencing in biblatex-chicago-notes then I strongly recommend you use Biber.

(One reason for this is that when Biber is the backend, biblatex defines a series of inheritance rules for the crossref field which make it much more convenient to use. Appendix B of biblatex.pdf explains the defaults, to which biblatex-chicago has added several that I should mention here: incollection entries can now inherit from book and mvbook just as they do from collection and mvcollection entries; letter entries now inherit from book, collection, mvbook, and mvcollection entries the same way an inbook or an incollection entry would; the namea, nameb, sortname, sorttitle, and sortyear fields, all highly single-entry specific, are no longer inheritable; and date and origdate fields are not inheritable from any of the new **mv*** entry types.)

Turning now to the provision of abbreviated references in biblatex-chicago-notes, the *Manual* (14.108) specifies that if you cite several contributions to the same collection, all (including the collection itself) may be listed separately in the bibliography, which the package does automatically, using the default inclusion threshold of 2 in the case both of crossref'ed and xref'ed entries. (The familiar \nocite command may also help in some circumstances.) In footnotes the specification suggests that, after a citation of any one contribution to the collection, all subsequent contributions may, even in the first, long footnote, be cited using a slightly shortened form, thus "avoiding clutter." In the bibliography the abbreviated form is appropriate for all the child entries. The biblatex-chicago-notes package has always implemented these instructions, but only if you use a crossref or an xref field, and only in incollection, inproceedings, or letter entries (on the last named, see just below). Recent releases have considerably extended this functionality.

First, I added five entry types — **book**, **bookinbook**, **collection**, **inbook**, **proceedings**, and **review** — to the list of those which use shortened cross references, and I added two

options — `longcrossref` and `booklongxref`, on which more below — which you can use in the preamble or in the `options` field of an entry to enable or disable the automatic provision of abbreviated references. (The `crossref` or `xref` field are still necessary for this provision, but they are no longer sufficient on their own.) The `inbook` and `review` types work exactly like `incollection` or `inproceedings`; in previous releases, you could use `inbook` instead of `incollection` to avoid the automatic abbreviation, the two types being otherwise identical. Now that you can use an option to turn off abbreviated references even in the presence of a `crossref` or `xref` field, I have thought it sensible to include this entry type alongside the others. (Cf. ellet:galena, keating:dearborn, lippincott:chicago, and prairie:state to see this mechanism in action in both notes and bibliography.) In the review type the mechanism is aimed primarily at blog comments, assuming you don't want to use the more convenient `commenton` `relatedtype`, which absolves you even of the need to provide a `title` field for such entries. See the documentation of the review type above for the details.

The inclusion of **book**, **bookinbook**, **collection**, and **proceedings** entries fulfills a request made by Kenneth L. Pearce, and allows you to obtain shortened references to, for example, separate volumes within a multi-volume work, or to different book-length works collected inside a single volume. Such references are not an explicit part of the *Manual's* specification, but they are a logical extension of it, so the system of options for turning on this functionality behaves differently for these four entry types than for the other 4 (see below). In notes-test.bib you can get a feel for how this works by looking at bernhard:boris, bernhard:ritter, bernhard:themacher, harley:ancient:cart, harley:cartography, and harley:hoc.

Before discussing the new package options, I should say a little about some subtleties involved in this mechanism. First, and especially for book, bookinbook, collection, and proceedings entries, it is much simpler if your backend is Biber, which allows you to provide `maintitle`s by cross-referencing an **mv*** entry, and `booktitle`s by cross-referencing book or collection entries. Second, where and when to print volume information in these references is extremely complex, and I confess that I designed the tests primarily with Biber in mind. Third, Andrew Goldstone long ago identified some other difficulties in the package's treatment of abbreviated citations, both in notes and bibliography, difficulties exacerbated now by the extension of the mechanism to book-like entries. If you refer separately to chapters in a single-author book, then the shortened part of the reference, to the whole book, won't repeat the author's name before the title of the whole. If, however, you refer separately to parts of a collection or proceedings, even when the editor of the collection is the same as the `author` of an essay in the collection, you will see the name repeated before the abbreviated part referencing the whole parent volume.

Shortened references to book-like entries require, I believe, a somewhat different treatment. Here, repeated `editor`s are avoided if the abbreviated reference is to a collection or proceedings entry, or to either of their mv* versions, while for other entry types repeated `author`s are avoided. Because the code in these situations tests for entry type, there may be corner cases where careful choice of the parent entry type gets you what you want. Likewise, judicious use of the `editor` and `editortype` fields may also help, in some circumstances, to clear names that are repeated unnecessarily. Also, because of the way dates are handled by the mv* entry types, and by child entries cross-referenced to such entry types, I thought it might help in these abbreviated book-like entries to provide a date for the `title` when it's part of a `maintitle`, though not when it's only part of a `booktitle`. If dates appear in shortened references where you'd rather not have them, I have provided the `omitxrefdate` option to turn them off, either in the preamble for the document as a whole or in the `options` field of individual entries. There is also an `xrefurl` option available to control the printing of `url`, `doi`, and `eprint` fields in abbreviated references where such information might otherwise never appear. See **mvbook** in section [4.1](#), and both `omitxrefdate` and `xrefurl` in section [4.4.3](#).

Finally, a published collection of letters also requires different treatment (14.111). If you cite more than one letter from the same collection, then the *Manual* specifies that only the collection itself should appear in the bibliography. In footnotes, you can use the `letter` entry type, documented above, for each individual letter, while the collection as a whole

may well require a book entry. I have, after some consideration, implemented the system of shortened references in letter entries, even though the *Manual* doesn't explicitly require it. (See white:ross:memo, white:russ, and white:total, for examples of the crossref field in action in this way, and please note that the second of these entries is entirely fictitious, provided merely for the sake of example.) How then to keep the individual letters from appearing in the bibliography? The simplest mechanism is probably just to use "skipbib" in the options field.

longcrossref   Returning, then, to the package options which control whether and where the abbreviated references appear, they function, by default, asymmetrically. The first, longcrossref, generally controls the settings for the entry types more-or-less authorized by the *Manual*: inbook, incollection, inproceedings, letter, and review.

> **false:** This is the default. If you use crossref or xref fields in the four mentioned entry types, you'll get the abbreviated references in both notes and bibliography.

> **true:** You'll get no abbreviated references in these entry types, either in notes or in the bibliography.

> **notes:** The abbreviated references will not appear in notes, but only in the bibliography.

> **bib:** The abbreviated references will not appear in the bibliography, but only in notes.

> **none:** This switch is special, allowing you with one setting to provide abbreviated references not just to the four entry types mentioned but also to book, bookinbook, collection, and proceedings entries, both in notes and in the bibliography.

booklongxref   The second option, booklongxref, controls the settings for book, bookinbook, collection, and proceedings entries:

> **true:** This is the default. If you use crossref or xref fields in these entry types, by default you will *not* get any abbreviated references, either in notes or bibliography.

> **false:** You'll get abbreviated references in these entry types both in notes and in the bibliography.

> **notes:** The abbreviated references will not appear in notes, but only in the bibliography.

> **bib:** The abbreviated references will not appear in the bibliography, but only in notes.

Please note that you can set both of these options either in the preamble or in the options field of individual entries, allowing you to change the settings on an entry-by-entry basis.

Please further note that in earlier releases of biblatex-chicago I recommended against using shorthand, reprinttitle and/or userf fields in combination with this abbreviated cross-referencing mechanism. I received, however, a request from Alexandre Roberts to allow the shorthand to appear in the place of the abbreviated cross-reference as an additional space-saving measure, and one from Kenneth Pearce to permit the combination of the other two fields with crossref, as well. The userf and reprinttitle fields should just work
inheritshorthand   automatically in such circumstances, but the shorthand field in parent entries needs to be enabled by setting the inheritshorthand package option to true. There are, in addition, several other steps required to make this function smoothly — please see the documentation of the **shorthand** field, below, for a full explanation. (In case it isn't clear, the combination of userf, shorthand, and crossref functionality in a single entry is now possible. If you come across any problems or inaccuracies, please report them.)

date   I have now implemented all of the applicable parts of biblatex's elegant, and long standing, support for the ISO8601-2 Extended Format specification, which means the package now provides greatly enhanced possibilities for presenting uncertain and unspecified dates and date ranges, along with date eras, seasons, and time stamps. I have also

implemented the *Manual's* (9.64) guidelines for compressing year ranges, as well as providing a few more extras to help with some of the other tricky corners of the *Manual's* instructions. A combination of biblatex and biblatex-chicago package options allows you to define when, how, and where any of these extended specifications will appear in your documents. I have attempted to provide as compliant a set of defaults as possible in biblatex-chicago.sty, but you can alter any of them according to your needs. All are documented in section 4.4, below, but table 3, located in the author-date section, purports to serve as a convenient reference guide to how this all works.

There are several more general remarks about the date field that may be helpful to users. First, I highly recommend familiarizing yourself with the extended date specifications, as in many cases they will greatly simplify the creation of your .bib databases. The new compressyears option (true by default), for example, takes a year range in a date field and handles the somewhat tricky Chicago compression rules for you, while also giving you a simple means of turning it off that doesn't involve combing your .bib file for all the year fields that contain your hand-formatted ranges. Clearly, situations may still arise when a specially-crafted year or origyear field may be necessary, but if you can use the enhanced specifications then I strongly advocate doing so. Second, the fine-grained specification of a time stamp is really only necessary for news stories that are frequently updated "as they unfold" (14.191), for online sources that change rapidly enough for a time stamp to be necessary (14.207, 14.233; wikiped:bibtex), or for online posts, particularly comments, that may need a time stamp for disambiguation (14.208–10). If you wish to specify the time zone, the *Manual* (10.41) prefers initialisms like "EST" or "PDT," and these are most easily provided using the timezone field, where you can include your own parentheses if so desired (cp. 14.191). For the date field itself, a time stamp will only appear in article, review, suppperiodical, and online entries, the first three only with a magazine entry-subtype. All types can print such a stamp from the urldate (controllable using the new urlstamp option), while only review and suppperiodical entries will print this data from an eventdate. If you find a context in which a time stamp would be useful and which isn't included in this discussion, please let me know.

urlstamp

Third, an incomplete time specification will be ignored by biber, so include the seconds in it, as in table 3, safe in the knowledge that they won't, by default, ever appear in your documents. Should you want that level of discrimination, the biblatex option seconds set to true provides it. Fourth, in the misc entry type the date field can help to distinguish between two classes of archival material, letters and "letter-like" sources using origdate while others (interviews, wills, contracts) use date. (See misc in section 4.1 for the details.) Fifth, you can in most entry types qualify a date with the userd field, assuming that the entry contains no urldate. For music and video entries, there are several other requirements — please see the documentation of userd, below.

Sixth, and finally, please note that the nameaddon field, which see, is no longer the place for time stamps, as it was in the 16th-edition styles. Any such data there should be moved into the corresponding date field (either the date or the eventdate, typically). On all these questions generally please cf. also origdate, timezone, and year, below; the alldates, all-times, alwaysrange, centuryrange, compressyears, datecirca, dateera, dateeraauto, dateuncertain, decaderange, nodatebrackets, nodates, noyearbrackets, timezones, urlstamp, and urltime options in sections 4.4.1, 4.4.2, and 4.4.3; and section 4.5.10 in biblatex.pdf

(Users of the Chicago author-date style who wish to minimize the labor needed to convert a .bib database for the notes & bibliography style should be aware that the latter style includes compatibility code for the cmsdate (silently ignored) and switchdates options, along with the mechanism for reversing date and origdate. This means that you can, in theory, leave all of this alone in your .bib file when making the conversion, though I'm retaining the right to revoke this if the code in question demonstrably interferes with the functioning of the notes & bibliography style.)

day This field, as of biblatex 0.9, is obsolete, and will be ignored if you use it in your .bib files. Use date instead.

doi Standard biblatex field, providing the Digital Object Identifier of the work. The *Manual* specifies that, given their relative permanence compared to URLs, "authors should prefer

a DOI- or Handle-based URL whenever one is available" (14.8). (14.175; friedman:learning). Cf. url.

edition Standard biblatex field. If you enter a plain cardinal number, biblatex will convert it to an ordinal (chicago:manual), followed by the appropriate string. Any other sort of edition information will be printed as is, though if your data begins with a word (or abbreviation) that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word (or abbreviation) is in lowercase, and biblatex-chicago-notes will automatically do the right thing (babb:peru, times:guide). In most situations, the *Manual* generally recommends the use of abbreviations in both bibliography and notes, but there is room for the user's discretion in specific citations (emerson:nature).

editor As far as possible, I have implemented this field as biblatex's standard styles do, but the requirements specified by the *Manual* present certain complications that need explaining. Biblatex.pdf points out that the editor field will be associated with a title, a booktitle, or a maintitle, depending on the sort of entry. More specifically, biblatex-chicago associates the editor with the most comprehensive of those titles, that is, maintitle if there is one, otherwise booktitle, otherwise title, if the other two are lacking. In a large number of cases, this is exactly the correct behavior (adorno:benj, centinel:letters, plato:republic:gr, among others). Predictably, however, there are numerous cases that require, for example, an additional editor for one part of a collection or for one volume of a multi-volume work. For these cases I have provided the namea field. You should format names for this field as you would for author or editor, and these names will always be associated with the title (donne:var).

As you will see below, I have also provided a nameb field, which holds the translator of a given title (euripides:orestes). If namea and nameb are the same, biblatex-chicago will concatenate them, just as biblatex already does for editor, translator, and namec (i.e., the compiler). Furthermore, it is conceivable that a given entry will need separate editors for each of the three sorts of title. For this, and for various other tricky situations, there is the \partedit macro (and its siblings), designed to be used in a note field or in one of the titleaddon fields (chaucer:liferecords). (Because the strings identifying an editor differ in notes and bibliography, one can't simply write them out in such a field, hence the need for a macro, which I discuss further in the commands section below [4.3.1].) Please note that, when attempting to find a name for the head of a note or a bibliography entry, namea takes precedence over editor, and nameb over translator. Cf. namea, nameb, namec, and translator.

editora The newer releases of biblatex provide these fields as a means to specify additional con-
editorb tributors to texts in a number of editorial roles. In the Chicago styles they seem most
editorc relevant for the audiovisual types, especially music and video, and now also the performance type, in all of which they can help to identify conductors, directors, producers, and performers. To specify the role, use the fields editoratype, editorbtype, and editorctype, which see. (Cf. bernstein:shostakovich, hamilton:miranda, handel:messiah.)

editortype Normally, with the exception of the article and review types, biblatex-chicago-notes will automatically find a name to put at the head of an entry, starting with an author, and proceeding in order through namea, editor, nameb, translator, and namec (the compiler). If all six are missing, then the title will be placed at the head. (In article and review entries with a magazine entrysubtype, a missing author immediately prompts the use of journaltitle at the head of an entry. See above under article for details.) The editortype field provides even greater flexibility, giving you the ability to indicate any number of roles at the head of an entry. You can do this even though an author is named (eliot:pound shows this mechanism in action for a standard editor, rather than for an alternative role). Two things are necessary for this to happen. First, in the options field you need to set use-author=false, then you need to put the name you wish to see at the head of your entry into the editor or the namea field. If the "editor" is in fact a compiler, then you need to put compiler into the editortype field, and biblatex will print the correct string after the name in both the bibliography and in the long note form.

In previous releases of biblatex-chicago you could only use defined `\bibstrings` in this field, at least if you wanted anything printed. N. Andrew Walsh pointed out that the standard biblatex styles will just print the field as-is in this case, allowing them to handle a great many unforeseen editorial roles with comparative ease, so I've implemented this, too, making sure to capitalize the string if the context demands it. The string you choose will differ depending on whether it will be printed after a name at the head of an entry or before a name later on in the entry, e.g., "cartographer" or "maps created by." A bit of trial and error should see you through.

There are a few more details of which you need to be aware. Because biblatex-chicago has added the namea field, which gives you the ability to identify the editor specifically of a title as opposed to a maintitle or a booktitle, the name-finding algorithm checks first to see whether a namea is defined. If it is, that name will be used at the head of the entry, if it isn't, or if you've set the option usenamea=false, the algorithm will go ahead and look for an editor. The editortype field applies only to the editor, but you can use nameatype to modify namea. Either of these names should be sorted properly in the bibliography, but please be aware that if you want a shortened form to appear in short notes then there's only the shorteditor, which you should ensure presents whichever of the two editors' names appears at the head of long notes or bibliography entries.

In biblatex 0.9 Lehman reworked the string concatenation mechanism, for reasons he outlined in his RELEASE file, and I have followed his lead. In short, if you define the editortype field, then concatenation is turned off, even if the name of the editor matches, for example, that of the translator. In the absence of an editortype (or nameatype), the usual mechanisms remain in place, that is, if the editor exactly matches a translator and/or a namec, or alternatively if namea exactly matches a nameb and/or a namec, then biblatex will print the appropriate strings. The *Manual* specifically (14.32) recommends not using these identifying strings in the short note form, and biblatex-chicago-notes follows their recommendation. If you nevertheless need to provide such a string, you'll have to do it manually in the shorteditor field, or perhaps, in a different sort of entry, in a shortauthor field.

It may also be worth noting that because of certain requirements in the specification – absence of an author, for example – the useauthor=false mechanism is either unnecessary or won't work properly in the following entry types: collection, letter, patent, periodical, proceedings, review, suppbook, suppcollection, and suppperiodical.

editoratype
editorbtype
editorctype
These fields identify the exact role of the person named in the corresponding editor[a-c] field, just as editortype (q.v.) does for the editor. Note that they are not part of the string concatenation mechanism. I have implemented them just as the standard styles do, that is, if the field isn't a pre-defined `\bibstring` it will be printed as-is, contextually capitalized. They have found a use particularly in music, performance, and video entries. Cf. bernstein:shostakovich, hamilton:miranda, handel:messiah.

eid
Standard biblatex field, providing a string or number some journals use uniquely to identify a particular article. Only applicable to the article entry type, and only to those without a magazine entrysubtype. The 17th edition of the *Manual* now specifies where to print this (14.174), and I have moved it in accordance with its specifications. It replaces the pages field in long notes and bibliography, and appears after any specific page cited in the postnote field of a long note.

entrysubtype
Standard and very powerful biblatex field, left undefined by the standard styles. In biblatex-chicago-notes it has eight very specific uses, the first three of which I have designed in order to maintain, as much as possible, backward compatibility with the standard styles. First, in article, periodical, and review entries, the field allows you to differentiate between scholarly "journals," on the one hand, and "magazines" and "newspapers" on the other. Usage is fairly simple: you need to put the exact string magazine into the entrysubtype field if you are citing one of the latter two types of source, whereas if your source is a "journal," then you need do nothing.

The second use involves references to works from classical antiquity and, according to the *Manual*, from the Middle Ages, as well. When you cite such a work using the tradi-

tional divisions into books, sections, lines, etc., divisions which are presumed to be the same across all editions, then you need to put the exact string classical into the entrysubtype field. This has no effect in long notes or in the bibliography, but it does affect the formatting of short notes, where it suppresses some of the punctuation. Ordinarily, you will use this toggle in a book or a bookinbook entry, but it is possible that a journal might well also present an edition of such a work. Given the tradition of using italics for the titles of such works, this may require using a titleaddon field (with hand formatting) instead of a title. If you wish to reference a classical or medieval work by the page numbers of a particular, non-standard edition, then you shouldn't use the entrysubtype toggle. Also, and the specification is reasonably clear about this, works from the Renaissance and later, even if cited by the traditional divisions, have short notes formatted normally, and therefore don't need an entrysubtype field. (See *Manual* 14.242–54; aristotle:metaphy:gr, herodotus:wilson, plato:republic:gr; euripides:orestes is an example of a translation cited by page number in a modern edition. Cf. also the notitle option in section 4.4.3.)

The third use occurs in misc entries. If such an entry contains no entrysubtype field, then the citation will be treated just as the standard biblatex styles would, including the use of italics for the title. Any string at all in entrysubtype tells biblatex-chicago-notes to treat the source as part of an unpublished archive. A misc entry with an entrysubtype defined is the least formatted of all those specified by the *Manual* — see section 4.1 above under **misc** for all the details on how these citations work.

Fourth, the field can be defined in the artwork entry type in order to refer to a work from antiquity whose title you do not wish to be italicized. Please see the documentation of **artwork** above for the details. Fifth, you can define it in a **standard** entry, q.v., to change the appearance of both long and short notes. Sixth, you can define it in an **audio**, **music**, or **video** entry if such an entry refers to an individual unit that isn't part of any larger collection, the entry therefore having only a title and not a booktitle, a title that biblatex-chicago would normally interpret as the title of a larger unit (and therefore italicize). Seventh, and sticking with the **video** type, though enacting quite a different syntactic transformation, the 17th edition (14.265) now recommends that, when presenting episodes from a TV series, the name of the series (booktitle) comes before the episode name (title). The exact string tvepisode in the entrysubtype field achieves this reversal, which includes using the booktitle as a sorttitle in the bibliography and also as the labeltitle in short notes.

Eighth, and finally, you can use any entrysubtype whatever in inreference entries in order to treat them as inherently online works rather than standard published works. See the documentation of **online** and **inreference** entries in section 4.1, above, and also 14.233 and wikiped:bibtex.

eprint Kazuo Teramoto suggested adding biblatex's excellent eprint handling to biblatex-chica-
eprintclass go, and he sent me a patch implementing it. I have applied it, with minor alterations, so
eprinttype these three fields now work more or less as they do in standard biblatex. They may prove helpful in providing more abbreviated references to online content than conventional URLs, though I can find no specific reference to them in the *Manual.*

eventdate This is a standard biblatex field which has gradually accumulated functions in biblatex-chicago. It can now play a role in artwork, audio, image, inproceedings, music, proceedings, review, standard, suppperiodical, unpublished, and video entries. In artwork and image entries it identifies the publication date of, most frequently, a photograph, in association with the howpublished field which identifies the periodical or other medium in which it was published (mccurry:afghangirl). In standard entries it will also usually be associated with a howpublished field, allowing you to specify a later renewal or reaffirmation of a standard (niso:bibref). In audio entries, it specifies the release date of a single episode of a podcast (danforth:podcast). In music entries, it identifies the recording or performance date of a particular song (rather than of a whole disc, for which you would use origdate), whereas in video entries it identifies either the original broadcast date of a particular episode of a TV series or the date of a filmed musical performance. In both these cases biblatex-chicago will automatically prepend a bibstring — recorded and aired, respectively — to the date, but you can change this string using the userd field, something you'll

definitely want to do for filmed musical performances (friends:leia, handel:messiah, holiday:fool).

In inproceedings, proceedings, and unpublished entries it identifies the date of an event at which a published or unpublished work was presented, though in truth the date will do as well in unpublished entries (nass:address). The field's use in review entries, finally, includes a possible time stamp. In this context, an eventdate helps to identify a particular comment on, or reply to another comment on, a blog post. Given that many such posts by a single author could appear on the same day, you can distinguish them by putting a time specification in the eventdate field itself (ac:comment). Please see the **review** type, above, for the details of how to cite these materials, possibly with the help of the commenton relatedtype. See also the date field docs, in particular table 3 (located in the author-date section), for details on how the ISO8601-2 Extended Format specifications offered by biblatex, including time stamps and much else besides, have been implemented in biblatex-chicago.

eventtimezone This field can, if necessary, specify the time zone associated with a time stamp given as part of an eventdate. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

eventtitle A standard biblatex field for identifying the name of the event that produces either a published record (inproceedings and proceedings entries) or an unpublished one (unpublished).

eventtitleaddon Standard biblatex field for adding information about an eventtitle, and available in the same entry types as that field.

foreword As with the afterword field above, foreword will in general function as it does in standard biblatex. Like afterword (and introduction), however, it has a special meaning in a suppbook entry, where you simply need to define it somehow (and leave afterword and introduction undefined) to make a foreword the focus of a citation.

holder A standard biblatex field for identifying a patent's holder(s), if they differ from the author. The *Manual* has nothing to say on the subject, but biblatex-chicago-notes prints it (them), in parentheses, just after the author(s).

howpublished Standard biblatex field which, like the eventdate field, is gradually accumulating functions in biblatex-chicago. In the booklet type it retains something of its traditional usage, replacing the publisher, and has a similar (somewhat paradoxical) place in unpublished entries. In the misc and performance types it works almost as a second note field, bringing in extra information about a work in close association with the type and version fields, while in dataset entries its information will be associated with both those fields and also with the number field. 17th-edition music entries require a field to provide the medium of downloaded music and/or the name of the streaming service, so howpublished works there as an online double of type and of publisher. Finally, in artwork, image, and standard entries it serves to qualify or modify an eventdate, almost as a userd field modifies a date or urldate. Please see the docs of those entry types for more information, and also bedford:photo, clark:mesopot, mccurry:afghangirl, niso:bibref, rihanna:umbrella.

institution Standard biblatex field. In the thesis entry type, it will usually identify the university for which the thesis was written, while in a report entry it may identify any sort of institution issuing the report.

introduction As with the afterword and foreword fields above, introduction will in general function as it does in standard biblatex. Like those fields, however, it has a special meaning in a suppbook entry, where you simply need to define it somehow (and leave afterword and foreword undefined) to make an introduction the focus of a citation.

isbn Standard biblatex field, for providing the International Standard Book Number of a publication. Not typically required by the *Manual*.

isrn Standard biblatex field, for providing the International Standard Technical Report Number of a report. Only relevant to the report entry type, and not typically required by the *Manual*.

issn Standard biblatex field, for providing the International Standard Serial Number of a periodical in an article or a periodical entry. Not typically required by the *Manual*.

issue Standard biblatex field, designed for article, periodical, or review entries identified by something like "Spring" or "Summer" rather than by the usual month or number fields (brown:bremer). Biblatex's enhanced date handling allows you to specify a season in the date field, with the "months" 21–24 used for Spring, Summer, Autumn, and Winter, respectively. Cf. table 3, below.

issuesubtitle The subtitle for an issuetitle — see next entry.

issuetitle Standard biblatex field, intended to contain the title of a special issue of any sort of periodical. If the reference is to one article within the special issue, then this field should be used in an article entry (conley:fifthgrade), whereas if you are citing the entire issue as a whole, then it would go in a periodical entry, instead (good:wholeissue). The note field is the proper place to identify the type of issue, e.g., special issue, with the initial letter lower-cased to enable automatic contextual capitalization.

journalsubtitle The subtitle for a journaltitle — see next entry.

journaltitle Standard biblatex field, replacing the standard BIBTEX field journal, which, however, still works as an alias. It contains the name of any sort of periodical publication, and is found in the article and review entry types. In the case where a piece in an article or review (entrysubtype magazine) doesn't have an author, biblatex-chicago-notes provides for this field to be used as the author. See above (section 4.1) under **article** for details. The lake-forester:pushcarts and nyt:trevorobit entries in notes-test.bib will give you some idea of how this works. Please note there is a shortjournal field which you can use to abbreviate the journaltitle in notes and/or in the bibliography, and you can also use it to print a list of journal abbreviations. Cf. the shortjournal documentation below.

journaltitleaddon An annex to the journaltitle, for which see previous entry. Such an annex would be printed in the main text font. If your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago-notes will automatically do the right thing. The package and entry option jtitleaddon (section 4.4.2) allows you to customize the punctuation that appears before the journaltitleaddon field (hua:cms). The default is a space.

keywords This field is biblatex's powerful and flexible technique for filtering bibliography entries, allowing you to subdivide a bibliography according to just about any criteria you care to invent, or indeed to prevent entries in notes from appearing in the bibliography, as the *Manual* sometimes recommends. See biblatex.pdf (3.7) for thorough documentation.

language A standard biblatex field, designed to allow you to specify the language(s) in which a work is written. As a general rule, the Chicago style doesn't require you to provide this information, though it may well be useful for clarifying the nature of certain works, such as bilingual editions, for example. There is at least one situation, however, when the *Manual* does specify this data, and that is when the title of a work is given in translation, even though no translation of the work has been published, something that might happen when a title is in a language deemed to be unparseable by a majority of your expected readership (14.99; pirumova, rozner:liberation). In such a case, you should provide the language(s) involved using this field, connecting multiple languages using the keyword and. (I have retained biblatex's \bibstring mechanism here, which means that you can use the standard bibstrings or, if one doesn't exist for the language you need, just give the name of the language, capitalized as it should appear in your text. You can also mix these two modes inside one entry without apparent harm.)

An alternative arrangement suggested by the *Manual* is to retain the original title of a piece but then to provide its translation, as well. If you choose this option, you'll need to make use of the **usere** field, on which see below. In effect, you'll probably only ever need to use one of these two fields in any given entry, and in fact biblatex-chicago-notes will only print one of them if both are present, preferring usere over language for this purpose (see kern and weresz). Note also that both of these fields are universally associated with the title of a work, rather than with a booktitle or a maintitle. If you need to attach a language or a translation to either of the latter two, you could probably manage it with special formatting inside those fields themselves.

lista I intend this field specifically for presenting citations from reference works that are arranged alphabetically, where the name of the item rather than a page or volume number should be given. The field is a biblatex list, which means you should separate multiple items with the keyword and. Each item receives its own set of quotation marks, and the whole list will be prefixed by the appropriate string ("s.v.," *sub verbo*, pl. "s.vv."). Biblatex-chicago-notes will only print such a field in a book or an inreference entry, and you should look at the documentation of these entry types for further details. (See *Manual* 14.232–33; ency:britannica, grove:sibelius, times:guide, wikiped:bibtex.)

location This is biblatex's version of the usual BIBTEX field address, though the latter is accepted as an alias if that simplifies the modification of older .bib files. According to the *Manual* (14.129), a citation usually need only provide the first city listed on any title page, though a list of cities separated by the keyword "and" will be formatted appropriately. If the place of publication is unknown, you can use \autocap{n}.p. instead (14.132). For all cities, you should use the common English version of the name, if such exists (14.131).

Three more details need explanation here. In article, periodical, and review entries, there is usually no need for a location field, but "if a journal might be confused with another with a similar title, or if it might not be known to the users of a bibliography," then this field can present the place or institution where it is published (14.182, 14.191, 14.193–94; lakeforester:pushcarts, kimluu:diethyl, and garrett). For blogs cited using article entries, this is a good place to identify the nature of the source — i.e., the word "blog" — letting the style automatically provide the parentheses (14.208; ellis:blog). Less predictably, it is in the vicinity of the location that the *Manual* indicates that a particular book is a reprint edition (14.114), so in such a case you can use the biblatex-chicago macro \reprint, followed by a comma, a space, and the location. Somewhat more cleanly and simply, and more in keeping with standard biblatex usage, you can just put the string reprint into the pubstate field to achieve the same result. See the pubstate documentation below (aristotle:metaphy:gr, schweitzer:bach). The origdate field may be used to give the original date of publication, and of course more complicated situations should usually be amenable to inclusion in the note field (emerson:nature).

mainsubtitle The subtitle for a maintitle — see next entry.

maintitle The main title for a multi-volume work, e.g., "Opera" or "Collected Works." (See donne: var, euripides:orestes, harley:cartography, lach:asia, pelikan:christian, and plato:republic:gr.) When using a crossref field and Biber, the title of **mv*** entry types always becomes a maintitle in the child entry. See also the documentation of the maintitle relatedtype in the **mvbook** docs in section 4.1, above, and in section 4.2.1, below.

Because the 17th edition of the *Manual* recommends that you present not only the names of blogs but also the names of their parent (usually periodical) publications, I have added this field to article, periodical, and review entries for just this purpose. See the documentation of those entry types in section 4.1, above, and also table 1 (14.208; amlen:hoot).

maintitleaddon An annex to the maintitle, for which see previous entry. Such an annex would be printed in the main text font. If your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago-notes will automatically do the right thing. The package and entry options ptitleaddon and ctitleaddon (section 4.4.2) allow you to customize the punctuation that appears before the maintitleaddon field (schubert:muellerin).

month Standard biblatex field, containing the month of publication. This should be an integer, i.e., month={3} not month={March}. See date for more information.

namea This is one of the fields biblatex provides for style writers to use, but which it leaves undefined itself. In biblatex-chicago it contains the name(s) of the editor(s) of a title, if the entry has a booktitle and/or a maintitle, in which situation the editor would be associated with one of these latter fields (donne:var). (In article and review entries, namea applies to the title instead of the issuetitle, should the latter be present.) You should present names in the field exactly as you would those in an author or editor field, and the package will concatenate this field with nameb if they are identical. When choosing a name to head a note or a bibliography entry, biblatex-chicago gives precedence to namea over editor. See under **editor** above for the full details. Please note that, as the field is highly single-entry specific, if you are using Biber namea isn't inherited from a crossref'ed parent entry. Please note, also, that you can use the nameatype field to redefine this role just as you can with editortype, which see. Cf. also nameb, namec, translator, and the macros \partedit, \parttrans, \parteditandtrans, \partcomp, \parteditandcomp, \parttransandcomp, and \partedittransandcomp, for which see section 4.3.1.

nameaddon This field is provided by biblatex, though not used by the standard styles. In biblatex-chicago its primary use, in most entry types, has always been to specify that an author's name is a pseudonym, or to provide either the real name or the pseudonym itself, if the other is being provided in the author field. The abbreviation "pseud." (always lower-case in English) is specified, either on its own or after the pseudonym (centinel:letters, creasey:ashe:blast, creasey:morton:hide, creasey:york:death, and lecarre:quest); remember that \bibstring{pseudonym} does the work for you. See under **author** above for the full details.

The field has slowly accumulated other functions, so when Philipp Immel made a feature request, and pointed to a discussion on Stack Exchange which suggested a few more, I thought I might generalize the field's functionality, providing three package options to allow users to mould it to their needs. Before discussing these, allow me to emphasize that the package defaults remain exactly the same as before, so that, absent any of the new options, the style still provides square brackets around the nameaddon in most entry types, no brackets of any sort in online, review, and suppperiodical entries, as well as in misc entries with an entrysubtype, and rather specialized handling in customc entries (which ignore all but the first of the new options — see below). If you're happy with the status quo, then no changes to your documents or .bib databases are necessary.

nameaddon   If you do need or want to put the field to a different use, the following options may help. All of them are available globally, per type, and per entry. The first new option is simply called nameaddon, and determines where and when the field will be printed at all. There are seven possible values:

> **all:** This is the default; if an entry has a nameaddon, it will appear in both long notes and in the bibliography.
> **none:** The field will appear neither in the bibliography nor in long notes.
> **bib:** The field will appear only in the bibliography.
> **cite:** The field will appear only in long notes.
> **first:** (This key and the next two are only available as global options.) Philipp Immel requested this as a way to provide an author's dates in the nameaddon field and only have them printed the first time that author appears in the bibliography. A sequence of consecutive long notes citing works by the same author will be treated the same way. The code tests for identical nameaddon fields in works by identical authors, so other sorts of nameaddon will be printed as usual.
> **bibfirst:** Like first, but will not print the nameaddon field in long notes.
> **citefirst:** Like first, but will not print the nameaddon field in the bibliography.

nameaddonsep   The nameaddonsep option controls the punctuation that appears before the nameaddon. It takes the following six keys:

> **space** = \addspace. This is the default.

none = no separator at all. It presumes that you will include one in the `nameaddon` field itself.

colon = \addcolon\addspace.

comma = \addcomma\addspace.

period = \addperiod\addspace.

semicolon = \addsemicolon\addspace.

nameaddon-format | The `nameaddonformat` option allows you to change the format of the `nameaddon` field on the fly, so its value should be a field format that biblatex understands. This includes standard formats like `parens`, `brackets` or `emph`, and also custom formats that you provide in your preamble using \DeclareFieldFormat, in case the standard ones aren't adequate. If you don't define this option, then the usual defaults apply, as delineated above, and you can use your own parentheses in online, review, and suppperiodical entries, as well as in misc entries with an `entrysubtype`, to distinguish screen names or other authorial information from traditional pseudonyms (in brackets).

Finally, two more details. If you are using the 17th-edition styles for the first time, please note that the 16th edition of the *Manual* recommended specifying comments to blogs and other online content using a time stamp in parentheses after the `author`, but the 17th edition handles time stamps both differently and more widely, so in this case you would now put time data into the `date` or `eventdate` field, particularly when the date itself is too coarse a specification to identify a comment unambiguously (cf. ac:comment, obrien:recycle). In the `customc` entry type, finally, which is used to create alphabetized cross-references to other bibliography entries, the `nameaddon` field allows you to change the default string linking the two parts of the cross-reference. The code automatically tests for a known bibstring, which it will italicize. Otherwise, it prints the string as you've provided in the `nameaddon` field itself. The punctuation is fixed.

nameatype | You can use this field to change the role of a `namea` just as you can use `editortype` to change the role of an `editor`. As with the `editortype`, using this field prevents string concatenation with identical `nameb` or `namec` fields. Please see **editortype**, above, for the details.

nameb | Like `namea`, above, this is a field left undefined by the standard biblatex styles. In biblatex-chicago, it contains the name(s) of the translator(s) of a title, if the entry has a `booktitle` or `maintitle`, or both, in which situation the translator would be associated with one of these latter fields (euripides:orestes). (In article and review entries, `nameb` applies to the title instead of the issuetitle, should the latter be present.) You should present names in this field exactly as you would those in an `author` or `translator` field, and the package will concatenate this field with `namea` if they are identical. See under the **translator** field below for the full details. Please note that, as the field is highly single-entry specific, if you are using Biber `nameb` isn't inherited from a crossref'ed parent entry. Please note, also, that in biblatex-chicago's name-finding algorithms `nameb` takes precedence over `translator`. Cf. `namea`, `namec`, origlanguage (section 4.2.1), translator, userf (section 4.2.1), and the macros \partedit, \parttrans, \parteditandtrans, \partcomp, \parteditandcomp, \parttransandcomp, and \partedittransandcomp in section 4.3.1.

namec | The *Manual* (14.103) specifies that works without an author may be listed under an editor, translator, or compiler, assuming that one is available, and it also specifies the strings to be used with the name(s) of compiler(s). All this suggests that the *Manual* considers this to be standard information that should be made available in a bibliographic reference, so I have added that possibility to the many that biblatex already provides, such as the editor, translator, commentator, annotator, and redactor, along with writers of an introduction, foreword, or afterword. Since biblatex doesn't offer a compiler field, I have adopted for this purpose the otherwise unused field `namec`. It is important to understand that, despite the analogous name, this field does not function like `namea` or `nameb`, but rather like editor or translator, and therefore if used will be associated with whichever title field these latter two would be were they present in the same entry. Identical fields among these three will be concatenated by the package, and concatenated too with the (usually) unnecessary commentator, annotator and the rest. Also please note that I've arranged the concatenation algorithms to include `namec` in the same test as `namea` and `nameb`, so

in this particular circumstance you can, if needed, make namec analogous to these two latter, title-only fields. (See above under **editortype** for details of how you may, in certain circumstances, use that field, or the nameatype field, to identify a compiler.)

It might conceivably be necessary at some point to identify the compiler(s) of a title separate from the compiler(s) of a booktitle or maintitle, but for the moment I've run out of available name fields, so you'll have to fall back on the \partcomp macro or the related \parteditandcomp, \parttransandcomp, and \partedittransandcomp, on which see Commands (section 4.3.1) below. (Future releases may be able to remedy this.) It may be as well to mention here too that of the names that can be substituted for the missing author at the head of an entry, biblatex-chicago-notes will choose a namea if present, then an editor, a nameb, or a translator, with namec coming last, assuming that the fields aren't identical, and therefore to be concatenated. The alphabetization routines should work properly for any of these names, but do please remember that if you want the package to skip over any names you can employ the use<name>=false options. Indeed, biblatex's usenamec has replaced the old Chicago-specific usecompiler, which is deprecated.

note   As in standard biblatex, this field allows you to provide bibliographic data that doesn't easily fit into any other field. In this sense, it's very like addendum, but the information provided here will be printed just before the publication data. (See chaucer:alt, chaucer:liferecords, cook:sotweed, emerson:nature, and rodman:walk for examples of this usage in action.) It also has a specialized use in all the periodical types (article, periodical, and review), where it holds supplemental information about a journaltitle, such as "special issue" (conley:fifthgrade, good:wholeissue). In all uses, if your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago-notes will automatically do the right thing. Cf. addendum.

number   This is a standard biblatex field, steadily accumulating uses in biblatex-chicago. It may contain the number of a journaltitle in an article or review entry, the number of a title in a periodical entry, the volume/number of a book (or musical recording) in a series, the (generally numerical) specifier of the type in a report entry, the archive location (or database accession number) of a dataset entry, and the number of a national or international standard in a standard entry. Generally, in an article, periodical, or review entry, this will be a plain cardinal number, but in such entries biblatex-chicago now does the right thing if you have a list or range of numbers (unsigned:ranke). In any book-like entry the field may well contain considerably more information, including even a reference to "2nd ser.," for example, while the series field in such an entry will contain the name of the series, rather than a number. This field is also the place for the patent number in a patent entry. Cf. issue and series. (Cf. 14.123–25 and boxer:china, palmatary:pottery, wauchope:ceramics; 14.171 and beattie:crime, conley:fifthgrade, friedman:learning, garrett, gibbard, hlatky:hrt, mcmillen:antebellum, rozner:liberation, and warr:ellison; 14.257 and genbank:db; 14.259 and niso:bibref; 14.263 and holiday:fool.)

NB: This may be an opportune place to point out that the *Manual* (14.147) prefers arabic to roman numerals in most circumstances (chapters, volumes, series numbers, etc.), even when such numbers might be roman in the work cited. The obvious exception is page numbers, in which roman numerals indicate that the citation came from the front matter, and should therefore be retained.

options   A standard biblatex field, for setting certain options on a per-entry basis rather than globally. Information about some of the more common options may be found above under author and below in section 4.4. See chaucer:alt, eliot:pound, herwign:office, lecarre:quest, and mla:style for examples of the field in use.

organization   A standard biblatex field, retained mainly for use in the misc, online, and manual entry types, where it may be of use to specify a publishing body that might not easily fit in other categories. In biblatex, it is also used to identify the organization sponsoring a conference in a proceedings or inproceedings entry, and I have retained this as a possibility, though the *Manual* is silent on the matter.

origdate This biblatex field allows you to provide more than one full date specification for those references which need it. As with the analogous date field, you provide the date (or range of dates) in ISO8601 format, i.e., yyyy-mm-dd. (You can also provide a time stamp in the field, after an uppercase "T", but I foresee this being very rarely needed in the notes & bibliography style. See table 3 for biblatex-chicago's implementation of biblatex's enhanced date specifications.) In most entry types, you would use origdate to provide the date of first publication of a work, most usually needed only in the case of reprint editions, but also recommended by the *Manual* for electronic editions of older works (14.114, 14.162; aristotle:metaphy:gr, emerson:nature, james:ambassadors, schweitzer:bach). In the letter and misc (with entrysubtype) entry types, the origdate identifies when a letter (or similar) was written. In such misc entries, you can choose between an origdate and a date field for this purpose, depending on how you want the date formatted (day-month-year or month-day-year, respectively), while in letter entries the date applies to the publication of the whole collection. If such a published collection were itself a reprint, improvisation in the location field might be able to rescue the situation. (See jackson:paulina: letter, white:ross:memo, white:russ, and white:total for how letter entries usually work; creel:house shows the field in action in a misc entry, while spock:interview uses date.)

In music entries, you can use the origdate in two separate but related ways. First, it can identify the recording date of an entire disc, rather than of one track on that disc, which would go in eventdate. (Compare holiday:fool with nytrumpet:art.) The style will automatically prepend the bibstring recorded to the date, but you can change it with the userd field. Be aware, however, that if an entry also has an eventdate, then userd will apply to that, instead, and you'll be forced to accept the default string. Second, the origdate can provide the original release date of an album. For this to happen, you need to put the string reprint in the pubstate field, which is a standard mechanism across many other entry types for identifying a reprinted work. (See floyd:atom.)

A couple of further notes are in order. First, artwork and image entries (which see) have their own scheme. Here, the style uses the earlier of two dates as the creation date of the work while the later is the printing date of, e.g., a particular exemplar of a photograph or of an etching. In such an entry, the origdate may well be a creation date. Second, because the origdate field only accepts numbers, some improvisation may be needed if you wish to include "n.d." (\bibstring{nodate}) in an entry. In letter and misc, this information can be placed in titleaddon, but in other entry types you may need to use the location field. (The origyear field usually works, too.)

origlanguage
origlocation See section 4.2.1, below.
origpublisher

origtimezone This field can, if necessary, specify the time zone associated with a time stamp given as part of an origdate. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

pages This is the standard biblatex field for providing page references. In many article and review entries you'll find this contains something other than a page number, e.g. a section name or edition specification (14.191; kozinn:review, nyt:obittrevor, nyt:trevorobit). Of course, the same may be true of almost any sort of entry, though perhaps with less frequency. Curious readers may wish to look at brown:bremer (14.180) for an example of a pages field used to facilitate reference to a two-part journal article. Cf. number for more information on the *Manual*'s preferences regarding the formatting of numerals; bookpagination and pagination provide details about biblatex's mechanisms for specifying what sort of division a given pages field contains; and usera discusses a different way to present the section information pertaining to a newspaper article.

David Gohlke brought to my attention a discussion that took place a couple of years ago on Stackexchange regarding the automatic compression of page ranges, e.g., 101--109 in the .bib file or in the postnote field would become 101–9 in the document. Biblatex has long had the facilities for providing this, and though the *Manual's* rules (9.61) are fairly complicated, Audrey Boruvka fortunately provided in that discussion code that implements the specifications. As some users may well be accustomed to compressing page ranges themselves in their .bib files, and in their postnote fields, I have made the activation of this

code a package option, so setting compresspages=true when loading biblatex-chicago should automatically give you the Chicago-recommended page ranges. **NB**: the code now resides in biblatex-chicago.sty, so if you don't load that package then you'll need to copy the code into your preamble for the option to have the desired effect.

pagination This, a standard biblatex field, allows you automatically to prefix the appropriate identifying string to information you provide in the postnote field of a citation command, whereas bookpagination allows you to prefix a string to the pages field. Please see **bookpagination** above for all the details on this functionality, as aside from the difference just mentioned the two fields are equivalent.

part Standard biblatex field, which identifies physical parts of a single logical volume in booklike entries, not in periodicals. It has the same purpose in biblatex-chicago-notes, but because the *Manual* (14.121) calls such a thing a "book" and not a "part," the string printed in notes and bibliography will, at least in English, be "bk." instead of the plain dot between volume number and part number (harley:cartography, lach:asia). If the field contains something other than a number, biblatex-chicago will print it as is, capitalizing it if necessary, rather than supplying the usual bibstring, so this provides a mechanism for altering the string to your liking. The field will be printed in the same place in any entry as would a volume number, and although it will most usually be associated with such a number, it can also function independently, allowing you to identify parts of works that don't fit into the standard scheme. If you need to identify "parts" or "books" that are part of a published series, for example, then you'll need to use a different field, (which in this case would be number [palmatary:pottery]). Cf. volume.

publisher Standard biblatex field. Remember that "and" is a keyword for connecting multiple publishers, so if a publisher's name contains "and," then you should either use the ampersand (&) or enclose the whole name in additional braces. (See *Manual* 14.133–41; aristotle:metaphy:gr, cohen:schiff, creasey:ashe:blast, dunn:revolutions.)

There are, as one might expect, a few further subtleties involved here. If you give two publishers in the field they will both be printed, separated by a forward slash in both notes and bibliography (14.90; sereny:cries). The 17th edition generally is rather keener than the 16th on using just one, particularly so in the case when the parent company of an imprint is also listed on a title page, in which case only the imprint need be included in your apparatus (14.138). If an academic publisher issues "certain books through a special publishing division or under a special imprint or as part of a publishing consortium (or joint imprint)," this arrangement may be specified in the publisher field (14.139; cohen:schiff). If a book has two co-publishers "in different countries" (14.140), then the simplest thing to do is to choose one, probably the nearest one geographically. If you feel it necessary to include both, then levistrauss:savage demonstrates one way of doing so, using a combination of the publisher and location fields. If the work is self-published, you can specify this in the pubstate field (see below), and any commercial self-publishing platform would go in publisher (14.137). Books published before 1900 can, at your discretion, include only the place (if known) and the date (14.128). If for some reason you need to indicate the absence of a publisher, the abbreviation given by the *Manual* is n.p., though this can also stand for "no place." The *Manual* also mentions s.n. (= *sine nomine*) to specify the lack of a publisher (10.42).

pubstate In response to new specifications in the 17th edition of the *Manual* (esp. 14.137), I have tried to generalize the functioning of the pubstate field in all entry types. The reprint string still has a special status there, being ignored in video entries and provoking a syntactic change in the presentation of dates in music entries, while in other types allowing the presentation of reprinted titles. Other strings are divided into two types: those which biblatex-chicago will print as the year, which currently means *only* those for which biblatex contains bibstrings indicating works soon to be published, i.e., forthcoming, in-preparation, inpress, and submitted; and those, i.e., everything else, which will be printed before, and in close association with, other information about the publisher of a work. The four in the first category will always be localized, as will reprint and self-published (and anything else that biblatex finds to be a \bibstring) from the second

category. All other strings will be printed as-is, capitalized if needed, just before the publisher (author:forthcoming, contrib:contrib, schweitzer:bach).

redactor I have implemented this field just as biblatex's standard styles do, even though the *Manual* doesn't actually mention it. It may be useful for some purposes. Cf. annotator and commentator.

reprinttitle See section 4.2.1, below.

series A standard biblatex field, usually just a number in an article, periodical, or review entry, almost always the name of a publication series in book-like entries, and providing similar identifying information associated with a number in music and standard entries. If you need to attach further information to the series name in a book-like entry, then the number field is again the place for it, whether it be a volume, a number, or even something like "2nd ser." or "\bibstring{oldseries}." Of course, you can also use \bibstring{oldseries} or \bibstring{newseries} in an article entry, but there you would place it in the series field itself. (In fact, the series field in article, periodical, and review entries is one of the places where biblatex allows you just to use the plain bibstring oldseries, for example, rather than making you type \bibstring{oldseries}. The type field in manual, patent, report, and thesis entries also has this auto-detection mechanism in place; see the discussion of \bibstring below for details.) In whatever entry type, these bibstrings produce the required abbreviation, which thankfully is the same in both notes and bibliography. (For books and similar entries, see 14.123–26; boxer:china, browning:aurora, palmatary:pottery, plato:republic:gr, wauchope:ceramics; for periodicals, see 14.184; garaud:gatine, sewall:letter; also niso:bibref, nytrumpet:art) Cf. number for more information on the *Manual*'s preferences regarding the formatting of numerals.

shortauthor This is a standard biblatex field, but biblatex-chicago-notes makes considerably greater use of it than the standard styles. For the purposes of the Chicago style, the field provides the name to be used in the short form of a footnote. In the vast majority of cases, you don't need to specify it, because the biblatex system selects the author's last name from the author field and uses it in such a reference, and if there is no author it will search namea, editor, nameb, translator, and namec, in that order. In an author-less article or review entry (entrysubtype magazine), where biblatex-chicago-notes will use the journaltitle as the author, you can use the shortjournal field instead, but you'll need to set up the journalabbrev option to make sure it's actually printed. (See shortjournal, below.) In author-less manual entries, where the organization will be so used, the style automatically uses any shortauthor in the short note form, though it will sort by the organization in the bibliography (dyna:browser, gourmet:052006, lakeforester:pushcarts, nyt:trevorobit).

As mentioned under editortype, the *Manual* (14.32) recommends against providing the identifying string (e.g., ed. or trans.) in the short note form, and biblatex-chicago-notes follows their recommendation. If you need to provide these strings in such a citation, then you'll have to do so by hand in the shortauthor field, or in the shorteditor field, whichever you are using.

shorteditor Like shortauthor, a field to provide a name for a short footnote, in this case for, e.g., a collection entry that typically lacks an author. The shortauthor field works just as well in most situations, but if you have set useauthor=false (and not useeditor=false) in an entry's options field, then only shorteditor will be recognized. It may be worth pointing out that, because biblatex-chicago also provides a namea field for the editor of a title as opposed to a main- or booktitle, and because in standard use the namea, if present, will be chosen to head a bibliography entry before the editor, you should present the shortened namea here instead of a shortened editor in such cases. Cf. editortype, above.

shorthand This is biblatex's mechanism for using abbreviations in place of the usual short note form, and in previous releases I left it effectively unmodified in biblatex-chicago-notes, apart from a few formatting tweaks. At the request of Kenneth Pearce, and following some hints in the *Manual*, I have made the system considerably more flexible, which I hope might be useful for those with specialized formatting needs. In the default configuration, any entry

which contains a shorthand field will produce a normal first note, either long or short according to your package options, informing the reader that the work will hereafter be cited by this abbreviation. As in standard biblatex, the \printshorthands command, now an alias for \printbiblist{shorthand}, will produce a formatted list of abbreviations for reference purposes, a list which the *Manual* suggests should be placed either in the front matter (when using footnotes) or before the endnotes, in case these are used.

I have provided five options to alter these defaults. First, there is a citation command, \shorthandcite, which will print the shorthand even at the first citation. I have only provided the most general form of this command, so you'll need to put it inside parentheses or in a \footnote command yourself. Alternately, you can set the shorthandfirst option to true, either in the preamble or in individual entries, to get the same effect, somewhat more simply. Third, I have included two bibenvironments for use with the env option to the \printshorthands command: losnotes is designed to allow a list of shorthands to appear inside footnotes, while losendnotes does the same for endnotes. Their main effect is to change the font size, and in the latter case to clear up some spurious punctuation and white space that I see on my system when using endnotes. (You'll probably also want to use the option heading=none in order to get rid of the [oversized] default, providing your own within the \footnote command.) Fourth, I have provided a package option, shorthandfull, which prints entries in the list of shorthands which contain full bibliographical information, effectively allowing you to eschew the bibliography in favor of a fortified shorthand list. (See 13.67, 14.59–60, and also biblatex.pdf for more information.) Finally, the shorthandintro option (section 4.4.3) allows you to control whether the first, long citation will introduce the shorthand at all, rather than just leaving it to the shorthand list (or perhaps "common knowledge") to clarify the reference.

Alexandre Roberts suggested a further refinement to shorthand behavior, which allows for it to appear in the place of the usual abbreviated citation of parent entries cross-referenced by several different child entries. In such a case, instead of the usual "... in Author, *Title*, 24–38," you would see instead "... in *ShrtHd*, 24–38." There are several steps required for enabling this behavior. First, you need to set the package option inherit-shorthand to true, which allows child entries to inherit the necessary fields from their cross-referenced parents. Second, you'll probably want to use the shorthandintro field somehow to clarify that the shorthand applies to the *parent* rather than to the *child*, as otherwise the reference will be ambiguous. Third, you'll need to put skipbiblist, formerly skiplos, in the options field of the child entries so that the shorthand itself appears in the list of shorthands *only* next to the parent entry, and not also next to all of its children.

As I mentioned above under **crossref**, I formerly recommended against using shorthands with cross-references, but this extension of their use makes sense as an extra space-saving measure. I'm not certain that I've identified all the possible drawbacks to enabling the inheritshorthand option, so care is still needed, at least in the current state of biblatex-chicago-notes. Please report any problems you might have with this functionality to the email address at the head of this documentation.

shorthandintro   When you include a shorthand in an entry, it will ordinarily appear the first time you cite the work, at the end of a long note, surrounded by parentheses and prefaced by the phrase "hereafter cited as." To modify this, you can either use the shorthandintro option (section 4.4.3) or you can use this standard biblatex field to change the formatting and the phrase to suit your needs. Please note, first, that you need to include the shorthand in this field as you intend it to appear and, second, that you still need the shorthand field present in order to ensure the appropriate presentation of that shorthand in later citations and in the list of shorthands. Finally, I've tried to allow for as many different styles of notification as possible, so by default the only punctuation that will appear between the rest of the citation and the shorthandintro is a space. You can change this punctuation, either in the preamble for the whole document or in individual entries, using the shorthand-punct option, documented in section 4.4.2. If the available option keys aren't adequate, you can use none and then provide custom punctuation inside the shorthandintro field itself.

shortjournal A special biblatex field, used both to provide an abbreviated form of a journaltitle in notes and/or bibliography and to facilitate the creation of a list of journal abbreviations rather in the manner of a shorthand list. As requested by user BenVB, you can now utilize this functionality in your documents, but there are several steps to take in order to do so. First, you'll need to provide both shortjournal and journaltitle fields in the entry types that use them, i.e., mainly article and review entries. In periodical entries the title field presents what would be the journaltitle in the previous two, so in such entries you can provide the standard shorttitle field to accompany the title, and biblatex-chicago will automatically copy the shorttitle into a shortjournal.

Having done this, you need to set the `journalabbrev` option either when loading biblatex-chicago or in the options field of individual .bib entries. By default, this option is not set, so your shortjournal fields will be silently ignored. There are three other settings: `true` prints the shortened fields both in notes and bibliography, `notes` only in notes, and `bib` only in the bibliography. Should you wish to present a list of these abbreviations with their expansions, then you need to use the `\printbiblist{shortjournal}` command, perhaps with a `title` option to differentiate the list from any shorthand list. As with shorthand lists, I have provided two `bibenvironments` for printing this list in foot- or endnotes (sjnotes and sjendnotes, respectively), to be used with the `env` option to `\printbiblist`. Again as with shorthands, you'll probably want to use the option `heading=none` when using these environments, just to turn off the (oversized) default, and perhaps provide your own title within the `\footnote` command. Finally, if you don't like the default formatting of the abbreviations in the list (bold italic), you can roll your own using `\DeclareFieldFormat{shortjournalwidth}` — you can see its default definition at the top of chicago-notes.bbx.

shortseries A special biblatex field, used both to provide an abbreviated form of a (book) series in notes and/or bibliography and to facilitate the creation of a list of such abbreviations rather in the manner of a shorthand list. As with the shortjournal field, its inclusion in biblatex-chicago was requested by user BenVB, and it is now available in entry types which have book-like series titles rather than journal-like numbers in the series field, to wit: audio, book, bookinbook, collection, inbook, incollection, inproceedings, inreference, letter, manual, music, mvbook, mvcollection, mvproceedings, mvreference, reference, report, standard, suppbook, and video. There are several steps to take in order to use the field. First, you'll need to provide both shortseries and series fields in the entry, then you'll need to set the `seriesabbrev` option either when loading biblatex-chicago, for the whole document or for specific entry types, or in the options field of individual .bib entries. By default, this option is not set, so your shortseries fields will be silently ignored. There are three other settings: `true` prints the shortened fields both in notes and bibliography, `notes` only in notes, and `bib` only in the bibliography. Should you wish to present a list of these abbreviations with their expansions, then you need to use the `\printbiblist{shortseries}` command, perhaps with a `title` option to differentiate the list from any shorthand list. As with shorthand lists, I have provided two `bibenvironments` for printing this list in foot- or endnotes (shsernotes and shserendnotes, respectively), to be used with the `env` option to `\printbiblist`. Again as with shorthands, you'll probably want to use the option `heading=none` when using these environments, just to turn off the (oversized) default, and perhaps provide your own title within the `\footnote` command. Finally, if you don't like the default formatting of the abbreviations in the list (plain roman), you can roll your own using `\DeclareFieldFormat{shortserieswidth}` — you can see its default definition at the top of chicago-notes.bbx.

shorttitle A standard biblatex field, primarily used to provide an abbreviated title for short notes. (It is also the way to hook periodical entries into the shortjournal mechanism, on which see the previous entry.) In biblatex-chicago-notes, you need to take particular care with letter entries, where, as explained above, the *Manual* requires a special format ("to Recipient"). (See 14.111; jackson:paulina:letter, white:ross:memo, white:russ.) Some misc entries (with an entrysubtype) also need special attention. (See creel:house, where the full title is used as the shortauthor + shorttitle by using `\headlesscite` commands.) Remember, also, that the generic titles in review and misc entries may not want capitalization in all contexts, so, as with the title field, if you begin a shorttitle with a lowercase

letter the style will do the right thing (barcott:review, bundy:macneil, Clemens:letter, kozinn:review, ratliff:review, unsigned:ranke).

sortkey Standard biblatex fields, designed to allow you to specify how you want an entry alphabet-
sortname ized in a bibliography. The sortkey field trumps all other sorting information, while the
sorttitle others offer more fine-grained control. In general, if an entry doesn't turn up where you
sortyear expect or want it, these fields should provide the solution. Entries with a corporate au-
thor can now omit the definite or indefinite article, which should help (14.70, 14.84; cot-
ton:manufacture, nytrumpet:art). Biblatex-chicago also includes the three supplemental
name fields (name[a-c]) in the sorting algorithm, so once again you should find that a
sortkey is needed less than before. Still, some entries without a name field of any sort,
particularly those with a definite or indefinite article beginning the title, may require as-
sistance (greek:filmstrip, grove:sibelius, nyt:obittrevor, virginia:plantation). Please con-
sult biblatex.pdf and the remarks on \DeclareSortingTemplate in section 4.4.1, below.

subtitle The subtitle for a title — see next entry.

timezone This field can, if necessary, specify the time zone associated with a time stamp given as
part of an date. The *Manual* prefers initialisms like "EST" for this purpose, and you can
provide parentheses around it at your discretion (cp. 10.41 and 14.191).

title In the vast majority of cases, this field works just as it always has in BIBTEX, and just as
it does in biblatex. Nearly every entry will have one, the most likely exceptions being
incollection or online entries with a merely generic title, instead of a specific one (cen-
tinel:letters, powell:email). The main source of difficulties flows from the *Manual*'s rules
for formatting titles, rules which also hold for booktitles and maintitles. The whole point
of using a biblatex-based system is for it to do the formatting for you, and in most cases
biblatex-chicago-notes does just that, surrounding titles with quotation marks, italicizing
them, or occasionally just leaving them alone. When, however, a title is quoted within a
title, then you need to know some of the rules. A summary here should serve to clarify
them, and help you to understand when biblatex-chicago-notes might need your help in
order to comply with them.

The internal rules of biblatex-chicago-notes are as follows:

Italics: booktitle, maintitle, and journaltitle in all entry types; title of artwork, book,
bookinbook, booklet, collection, image, manual, misc (with no entrysubtype), per-
formance, periodical, proceedings, report, standard, suppbook, and suppcollection
entry types.

Quotation Marks: title of article, inbook, incollection, inproceedings, online, period-
ical, thesis, and unpublished entry types, issuetitle in article, periodical, and review
entry types.

Sentence cased: title in patent entries.

Unformatted: booktitleaddon, maintitleaddon, and titleaddon in all entry types, title
of customc, letter, misc (with an entrysubtype), review, and suppperiodical entry
types.

Italics or Quotation Marks: All of the audiovisual entry types — audio, music, and
video — have to serve as analogues both to book and to inbook. Therefore, if there
is both a title and a booktitle, then the title will be in quotation marks. If there is no
booktitle, then the title will be italicized, unless you provide an entrysubtype.

Now, the rules for which entry type to use for which sort of work tend to be fairly straight-
forward, but in cases of doubt you can consult section 4.1 above, the examples in notes-
test.bib, or go to the *Manual* itself, 8.156–201. Assuming, then, that you want to present
a title within a title, and you know what sort of formatting each of the two would, on its
own, require, then the following rules apply:

1. Inside an italicized title, all other titles are enclosed in quotation marks and italicized, so in such cases all you need to do is provide the quotation marks using \mkbibquote, which will take care of any following punctuation that needs to be brought within the closing quotation mark(s) (14.94; donne:var, mchugh:wake).

2. Inside a quoted title, you should present another title as it would appear if it were on its own, so in such cases you'll need to do the formatting yourself. Within the double quotes of the title another quoted title would take single quotes — the \mkbibquote command does this for you automatically, and also, I repeat, takes care of any following punctuation that needs to be brought within the closing quotation mark(s). (See 14.94–95; garrett, loften:hamlet, murphy:silent, white:callimachus.)

3. Inside a plain title (most likely in a review entry or a titleaddon field), you should present another title as it would appear on its own, once again formatting it yourself using \mkbibemph or \mkbibquote. (barcott:review, gibbard, osborne:poison, ratliff:review, unsigned:ranke).

The *Manual* provides a few more rules, as well. A word normally italicized in text should also be italicized in a quoted or plain-text title, but should be in roman ("reverse italics") in an italicized title. A quotation used as a (whole) title (with or without a subtitle) retains, according to the 16th edition, its quotation marks in an italicized title if it appears that way in the source, but I can't find similar instructions in the 17th. Such a quotation always retains its quotation marks when the surrounding title is quoted or plain (14.94; lewis). A word or phrase in quotation marks, but that isn't a quotation, retains those marks in all title types (kimluu:diethyl).

Finally, please note that in all review (and suppperiodical) entries, and in misc entries with an entrysubtype, and only in those entries, biblatex-chicago-notes will automatically capitalize the first word of the title after sentence-ending punctuation, assuming that such a title begins with a lowercase letter in your .bib database. See **\autocap** in section 4.3.1 below for more details.

titleaddon   Standard biblatex intends this field for use with additions to titles that may need to be formatted differently from the titles themselves, and biblatex-chicago-notes uses it in just this way, with the additional wrinkle that it can, if needed, replace the title entirely, and this in, effectively, any entry type, providing a fairly powerful, if somewhat complicated, tool for getting biblatex to do what you want (cf. centinel:letters, powell:email). This field will always be unformatted, that is, neither italicized nor placed within quotation marks, so any formatting you may need within it you'll need to provide manually yourself. The single exception to this rule is when your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, in which case you need then simply ensure that that word is in lowercase, and biblatex-chicago-notes will automatically do the right thing. See **\autocap** in section 4.3.1, below. The package and entry options ptitleaddon and ctitleaddon (section 4.4.2) allow you to customize the punctuation that appears before the titleaddon field. Please note, however, that I have added this field to the periodical entry type, and that the punctuation there is governed by the jtitleaddon option, which defaults to a space. (Cf. brown:bremer, osborne:poison, reaves:rosen, and white:ross:memo for examples where the field starts with a lowercase letter; morgenson:market provides an example where the titleaddon field, holding the name of a regular column in a newspaper, is capitalized, a situation that is handled as you would expect; coolidge:speech shows both entry options for controlling the punctuation.)

translator   As far as possible, I have implemented this field as biblatex's standard styles do, but the requirements specified by the *Manual* present certain complications that need explaining. Biblatex.pdf points out that the translator field will be associated with a title, a booktitle, or a maintitle, depending on the sort of entry. More specifically, biblatex-chicago associates the translator with the most comprehensive of those titles, that is, maintitle if there is one, otherwise booktitle, otherwise title, if the other two are lacking. In a large number of cases, this is exactly the correct behavior (adorno:benj, centinel:letters, plato:republic:gr, among others). Predictably, however, there are numerous cases that require, for example, an additional translator for one part of a collection or for one volume of a multi-volume

work. For these cases I have provided the nameb field. You should format names for this field as you would for author or editor, and these names will always be associated with the title (euripides:orestes) In the algorithm for finding a name for the head of notes and bibliography entries, nameb takes precedence over translator.

I have also provided a namea field, which holds the editor of a given title (euripides:orestes). If namea and nameb are the same, biblatex-chicago will concatenate them, just as biblatex already does for editor, translator, and namec (i.e., the compiler). Furthermore, it is conceivable that a given entry will need separate translators for each of the three sorts of title. For this, and for various other tricky situations, there is the \parttrans macro (and its siblings), designed to be used in a note field or in one of the titleaddon fields (ratliff:review). (Because the strings identifying a translator differ in notes and bibliography, one can't simply write them out in such a field, hence the need for a macro, which I discuss further in the commands section below [4.3.1].)

Finally, as I detailed above under **author**, in the absence of an author, namea, editor, and nameb, the translator will be used at the head of an entry (silver:gawain), and the bibliography entry alphabetized by the translator's name, behavior that can be controlled with the use<name> switches in the options field. Cf. author, editor, namea, nameb, and namec.

type This is a standard biblatex field, and in its normal usage serves to identify the type of a manual, patent, report, or thesis entry. Biblatex 0.7 introduced the ability, in some circumstances, to use a bibstring without inserting it in a \bibstring command, and in some entry types (audio, manual, music, patent, report, suppbook, suppcollection, thesis, and video) the type field works this way, allowing you simply to input, e.g., patentus rather than \bibstring{patentus}, though both will work. (See petroff:impurity; herwign:office, murphy:silent, and ross:thesis all demonstrate how the type field may sometimes be automatically set in such entries by using one of the standard entry-type aliases). In other entry types (artwork, image, book, online, article, review, and suppperiodical) biblatex-chicago will merely capitalize the contents according to context.

In the suppbook entry type, and in its alias suppcollection, you can use the type field to specify what sort of supplemental material you are citing, e.g., "preface to" or "postscript to." Cf. suppbook above for the details. (See *Manual* 14.110; polakow:afterw, prose:intro).

You can use the type field in artwork, audio, image, music, and video entries to identify the medium of the work, e.g., oil on canvas, albumen print, compact disc or MPEG. In book entries it will normally hold system information about multimedia app content (14.268), while in online, article, and review entries it will hold the medium of online multimedia (14.267). Cf. under these entry types in section 4.1, above, for more details. (See auden:reading, bedford:photo, cleese:holygrail, leo:madonna, nytrumpet:art.)

url A standard biblatex field, it holds the url of an online publication, though you can provide one for all entry types. The *Manual* expresses a strong preference for DOIs over URLs if the former is available — cf. doi above, and also urldate just below. The required LaTeX package url will ensure that your documents format such references properly, in the text and in the reference apparatus. It may be worth noting that child entries generally won't inherit url fields from their parents — the information seems entry-specific enough to warrant a little bit of extra typing if you need to present the same locator in several entries. You
blogurl can, however, set the preamble option blogurl to allow your child comments (review) to inherit the URL from the parent blog (article).

urldate A standard biblatex field, it identifies exactly when you accessed a given url, and is given in ISO8601 format. The *Manual* prefers DOIs to URLs; in the latter case it allows the use of access dates, particularly in contexts that require it, but prefers that you use revision dates, if these are available. To enable you to specify which date is at stake, I have provided the **userd** field, documented below. If an entry doesn't have a userd, then the urldate will be treated as an access date (14.8, 14.12–13, 14.207; evanston:library, grove:sibelius, hlatky:hrt, osborne:poison, sirosh:visualcortex, wikiped:bibtex). You can also use the

field to specify a time stamp, should the date alone not be specific enough. The time stamp follows the date, separated by an uppercase "T", like so: yyyy-mm-ddThh:mm:ss. If you wish to specify the time zone, the *Manual* (10.41) prefers initialisms like "EST" or "PDT," and these are most easily provided using the urltimezone field, where you can provide your own parentheses if so desired (cp. 14.191). Following the examples in the *Manual*, any urldate will by default be printed in 24-hour format, though other time stamps use 12-hour format. The biblatex option urltime, discussed in section 4.4.1, allows you to change this in your preamble.

urlstamp

A urldate time stamp (and urltimezone) can appear in any entry whatsoever, if you judge the online source to be the sort that changes rapidly enough for a time stamp to be necessary (14.207, 14.233; wikiped:bibtex). You can stop it printing by setting the new urlstamp option to false in your preamble for the whole document or for specified entry types, or in the options field of individual entries. Please see the documentation of **date**, above, and also table 3, below, for more details about time stamps and other parts of biblatex's enhanced date specifications. Table 1 contains a summary of the current state of biblatex-chicago's handling of online materials.

urltimezone

This field can, if necessary, specify the time zone associated with a time stamp given as part of an urldate. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

usera

A supplemental biblatex field which in certain contexts in biblatex-chicago will identify the broadcast network when you cite a radio or television program. In article, periodical, and review entries with entrysubtype magazine, it acts almost as a "journaltitleaddon" field, and its contents will be placed, unformatted and between commas, after the journaltitle and before the date. In video entries it comes after the eventdate, i.e., the date of first broadcast, and is separated from that date by the \bibstring "on" (14.213, 14.265; american:crime, bundy:macneil, friends:leia, mayberry:brady).

userc

I have implemented this supplemental biblatex field as part of Chicago's name cross-referencing system. (The "c" part is meant as a sort of mnemonic for this function, though it's perfectly possible to use the field in other contexts.) If you use the **customc** entry type to include alphabetized cross-references to other, separate entries in a bibliography, it is unlikely that you will cite the customc entry in the body of your text. Therefore, in order for it to appear in the bibliography, you have two choices. You can either include the entry key of the customc entry in a \nocite command inside your document, or you can place that entry key in the userc field of another .bib entry that you will be citing. In the latter case, biblatex-chicago will call \nocite for you, and this method should ensure that there will be at least one entry in the bibliography to which the cross-reference will point. (See 14.81–82; creasey:ashe:blast, creasey:morton:hide, creasey:york:death, lecarre:quest.)

userd

The userd field acts as a sort of "datetype" field, allowing you in most entry types to identify whether a urldate is an access date or a revision date. The general usage is fairly simple. If this field is absent, then a urldate will be treated as an access date, as has long been the default in biblatex and in biblatex-chicago. If you need to identify it in any other way, what you include in userd will be printed *before* the urldate, so phrases like "last modified" or "last revised" are what the field will typically contain (14.12–13; wikiped:bibtex). In the absence of a urldate you can, in most entry types, include a userd field to qualify a date in the same way it would have modified a urldate.

Because of the rather specialized needs of some audio-visual references, this basic schema changes for music and video entries. In music entries where an eventdate is present, userd will modify that date instead of any urldate that may also be present, and it will modify an origdate if it is present and there is no eventdate. It will modify a date only in the absence of the other three. In video entries it will modify an eventdate if it is present, and in its absence the urldate. In the absence of those two, it can modify a date. Please see the documentation of the **music** and **video** entry types, and especially of the eventdate, origdate, and urldate fields, above (14.263–65; nytrumpet:art).

In all cases, you can start the userd field with a lowercase letter, and biblatex will take care of automatic contextual capitalization for you.

usere Another supplemental biblatex field, which biblatex-chicago uses specifically to provide a translated title of a work, something that may be needed if you deem the original language unparseable by a significant portion of your likely readership. The *Manual* offers two alternatives in such a situation: either you can translate the title and use that translation in your title field, providing the original language in language, or you can give the original title in title and the translation in usere. If you choose the latter, you may need to provide a shorttitle so that the short note form is also parseable. Cf. **language**, above. (See 14.99; kern, weresz.)

userf See section 4.2.1, below.

venue Standard biblatex offers this field for use in proceedings and inproceedings entries, and after a request from Patrick Danilevici I have followed suit. I have also implemented the field in the misc entry type, both with and without an entrysubtype, in the performance type, and in the unpublished type. In all uses it will normally present the actual venue of an event, as opposed, e.g., to the origlocation, which might present where a letter was written or where an earlier edition was printed.

version Standard biblatex field, formerly only available in artwork, image, misc, music, and patent entries in biblatex-chicago-notes, but now also in book and performance entries. In most entry types it prints a localized "version" string, but there may be specialist needs in artwork and image entries, so there you'll need to specify the type of data inside the field itself. In the book type it is particularly needed for presenting multimedia app content (14.268).

volume Standard biblatex field. It holds the volume of a journaltitle in article (and some review) entries, and also the volume of a multi-volume work in many other sorts of entry. The treatment and placement of volume information in book-like entries is rather complicated in the *Manual* (14.116–22). In bibliography entries, the volume appears either before the maintitle or before the publication information. In long notes, the same applies, but with the additional possibility of this information appearing *after* the publication data, just before page numbers. In the past, if you wanted the volume information to appear here, you had to leave that information out of your .bib entry and give it in the pages or
delayvolume postnote field. Now, you can use the biblatex-chicago option delayvolume in your preamble or in the options field of an entry to ensure that any volume information that would normally have appeared just before the publication data in a long note appears after it.

The volume information in both books and periodicals, and in both the bibliography and long notes, can appear *immediately before* the page number(s). In such a case, the *Manual* prescribes the same treatment for both sorts of sources, that is, that "a colon separates the volume number from the page number with no intervening space." I have implemented this, but at the request of Clea F. Rees I have made this punctuation customiz-
\postvolpunct able, using the command \postvolpunct. By default it prints \addcolon, but you can use \renewcommand{\postvolpunct}{…} in your preamble to redefine it. Cf. part, and the command documentation in section 4.3.1.

volumes Standard biblatex field. It holds the total number of volumes of a multi-volume work (meredith:letters). If both a volume and a volumes field are present, as may occur particularly in cross-referenced entries or in entries using the maintitle relatedtype, then biblatex-chicago will ordinarily suppress the volumes field in your references, except in some instances when a maintitle is present. If the volume appears before the maintitle,
hidevolumes the option hidevolumes, set to true by default, controls whether to print the volumes field after that title or not. If it appears after the maintitle, as with the relatedtype just mentioned, the same option controls whether to print the volumes field in close association with the volume. Set the option to false either in the preamble or in the options field of your entry to have the volumes appear in these circumstances. Cf. the option's documentation in section 4.4.2, below.

xref A modified `crossref` field provided by `biblatex`, which prevents inheritance of any data from the parent entry. See **crossref**, above.

year Standard `biblatex` field. It usually identifies the year of publication, though unlike the date field it allows non-numeric input, so you can put "n.d." (or, to be language agnostic, `\bibstring{nodate}`) here if required, or indeed any other sort of non-numerical date information. For many kinds of uncertain and unspecified dates it is now much simpler to make use of `biblatex`'s enhanced date specifications in the date field, instead. Please see table 3 for a summary of how `biblatex-chicago` implements these enhancements. Cf. bedford:photo, clark:mesopot, leo:madonna, ross:thesis.

4.2.1   Fields for Related Entries

As biblatex.pdf puts it (§ 3.4), "Almost all bibliography styles require authors to specify certain types of relationship between entries such as 'Reprint of', 'Reprinted in,' etc. It is impossible to provide data fields to cover all of these relationships and so `biblatex` provides a general mechanism for this using the entry fields related, relatedtype and relatedstring." Before this mechanism was available `biblatex-chicago` attempted to provide a similar but much more limited set of inter-entry relationships using the `biblatex` fields origlanguage, origlocation, origpublisher, pubstate, reprinttitle, and userf. All of these still work just as they always have or, I hope, somewhat better than they always have after many recent bug fixes, but the more general and more powerful `biblatex` related mechanism is also available. It can provide much of what the older system provided and a great deal that it couldn't. What follows is a field-by-field discussion of the options now available.

origlanguage In keeping with the *Manual*'s specifications, I have fairly thoroughly redefined `biblatex`'s facilities for treating translations. The origtitle field isn't used, while the language and origdate fields have been press-ganged for other duties. The origlanguage field, for its part, retains a dual role in presenting translations in a bibliography. The details of the *Manual*'s suggested treatment when both a translation and an original are cited may be found below under **userf**. Here, however, I simply note that the introductory string used to connect the translation's citation with the original's is "Originally published as," which I suggest may well be inaccurate in a great many cases, as for instance when citing a work from classical antiquity, which will most certainly not "originally" have been published in the Loeb Classical Library. Although not, strictly speaking, authorized by the *Manual*, I have provided another way to introduce the original text, using the origlanguage field, which must be provided *in the entry for the translation, not the original text* (aristotle:metaphy:trans). If you put one of the standard `biblatex` bibstrings there (enumerated below), then the entry will work properly across multiple languages. Otherwise, just put the name of the language there, localized as necessary, and `biblatex-chicago` will eschew "Originally published as" in favor of, e.g., "Greek edition:" or "French edition:". This has no effect in notes, where only the work cited — original or translation — will be printed, but it may help to make the *Manual*'s suggestions for the bibliography more palatable. **NB:** You can use the relatedtype origpubas with a customized relatedstring field to achieve the same ends.

That was the first usage, in keeping at least with the spirit of the *Manual.* I have also, perhaps less in keeping with that specification, retained some of `biblatex`'s functionality for this field. If an entry doesn't have a userf field, and therefore won't be combining a text and its translation in the bibliography, you can also use origlanguage as the standard styles use it, so that instead of saying, e.g., "translated by X," the entry will read "translated from the German by X." The *Manual* doesn't mention this, but it may conceivably help avoid certain ambiguities in some citations. As in `biblatex`, if you wish to use this functionality, you have to provide *not* the name of the language, but rather a bibliography string, which may, at the time of writing, be one of `american`, `brazilian`, `danish`, `dutch`, `english`, `french`, `german`, `greek`, `italian`, `latin`, `norwegian`, `portuguese`, `spanish`, or `swedish`, to which I've added `russian`.

origlocation This field mainly serves to help document reprint editions and their corresponding originals (14.114). In biblatex-chicago you can provide both an origlocation and an origpublisher to go along with the origdate, should you so wish, and all of this information will be

54

printed in long notes and bibliography. You can also use this field in a letter or misc (with entrysubtype) entry to give the place where a published or unpublished letter was written (14.111, 14.229). (Jonathan Robinson has suggested that the origlocation may in some circumstances actually be necessary for disambiguation, his example being early printed editions of the same material printed in the same year but in different cities. The new functionality should make this simple to achieve. Cf. origdate (section 4.2), origpublisher and pubstate; schweitzer:bach.) **NB:** It is impossible to present this same information, as here, *inside* a single entry using a related field, though the relatedtype origpubin presents much the same information *after* the entry, using data extracted from a separate entry.

origpublisher   As with the origlocation field just above, this field mainly serves to help document reprint editions and their corresponding originals (14.114). You can provide an origpublisher and/or an origlocation in addition to the origdate, and all will be presented in long notes and bibliography. (Cf. origdate (section 4.2), origlocation, and pubstate; schweitzer:bach.) **NB:** It is impossible to present this same information, as here, *inside* a single entry using a related field, though the relatedtype origpubin presents much the same information *after* the entry, using data extracted from a separate entry.

pubstate   In response to new specifications in the 17th edition of the *Manual* (esp. 14.137), I have tried to generalize the functioning of the pubstate field in all entry types. The reprint string still has a special status there, being ignored in video entries and provoking a syntactic change in the presentation of dates in music entries (14.263; floyd:atom), while in other types allowing the presentation of reprinted titles. Other strings are divided into two types: those which biblatex-chicago will print as the year, which currently means *only* those for which biblatex contains bibstrings indicating works soon to be published, i.e., forthcoming, inpreparation, inpress, and submitted; and those, i.e., everything else, which will be printed before, and in close association with, other information about the publisher of a work. The four in the first category will always be localized, as will reprint and selfpublished (and anything else that biblatex finds to be a \bibstring) from the second category. All other strings will be printed as-is, capitalized if needed, just before the publisher (author:forthcoming, contrib:contrib, schweitzer:bach). **NB:** The pubstate functionality currently has no equivalent using the related field.

related   This field is required to use biblatex's related functionality, and it should contain the entry key or keys from which biblatex should extract data for presentation not on its own, but rather in the bibliography entry (or long note) which contains the related field itself. Indeed, unless you change the defaults using the relatedoptions field this data will only appear in such entries, never on its own. Without a relatedtype field, this will print the default type, equivalent to a long note citation *immediately after* the bibliography entry containing the related field, with no intervening string. You can specify a string using the relatedstring field, so in effect this presents a powerful mechanism for presenting full references to related material of any sort whatsoever.

related=bib   By default, the package option related is set to print related entries only in the bibliography. If you would like them to appear only in long notes, in both notes and bibliography, or indeed in neither, you can set this option, either in your preamble (globally or for specific entry types) or in the options field of the relevant entry, to notes, true, or false, respectively (coolidge:speech and weed:flatiron). For the three relatedtypes that construct a single entry using data extracted from related entries — commenton, maintitle, and reviewof — biblatex-chicago will automatically set it to true for you entry by entry, as this is required to get properly-formatted citations in notes and bibliography. See below for the details.

relatedoptions   This field will, I should expect, only be needed very rarely. If you want to set entry-level options for a related entry this is where you can do it, though please remember one important detail. By default, Biber sets this option to dataonly, which among other things prevents the related entry from appearing separately in the bibliography. If you use the field yourself, then you'll need to include dataonly as one of the options therein to maintain this effect. Of course, it may be you don't want all the effects of dataonly, so you can tailor it however you wish. See biblatex.pdf § 3.4.

relatedstring The procedure for choosing a string to connect the main entry with its related entry/ies is straightforward, the default being a bibstring, if any, with the same name as the relatedtype, or alternately a string or strings defined within the driver for that relatedtype, as happens with the types origpubin and bytranslator. Failing these, you can supply your own in the relatedstring field, either in the form of the name of a pre-defined bibstring or as any text you choose, and anything in this field always takes precedence over the automatic choices. If your non-bibstring starts with a lowercase letter then biblatex-chicago will capitalize it automatically for you depending on context (weed:flatiron). I have not altered the standard relatedtype strings, and have indeed changed the reprint-title mechanism to use the reprintfrom string, which works better syntactically in this context.

relatedtype The standard biblatex styles define six relatedtypes, and I have either simply adopted them wholesale or adapted them to the needs of the Chicago style, retaining the basic syntax as much as possible. I have also added four to these six (see below). First, the standard types:

> **bytranslator:** This prints a full citation of a translation, starting with the (localized) string "Translated by translator as Title, …" The reference is fuller in biblatex-chicago than in the standard styles, and for the first time allows users to choose the *Manual's* alternate method for presenting original + translation (14.99; furet:related). The old userf mechanism provides the other, as does the origpubas relatedtype (see below).

> **default:** This is the macro used when no relatedtype is defined. It prints, as in the standard styles, and with no intervening string, a full citation of related entries. In biblatex-chicago-notes, the citation is in long note form, rather than bibliography form, as this is the usual practice in the *Manual.*

> **multivolume:** This briefly lists the individual volumes in a multi-volume work, and works much as in the standard styles. The *Manual*, as far as I can see, has little to say on the matter.

> **origpubas:** This type can, if you want, replace the old userf mechanism, described below, for presenting an original with its translation. It's quite similar to the default type, but with a bibstring automatically connecting the entry with its related entries. You can identify other sorts of relationships if you change the introductory string using relatedstring.

> **origpubin:** I haven't altered this from the biblatex default at all, and it presents reprint information *after* the main entry rather than within it. The *Manual* seems to prefer the latter for the notes & bibliography style and, in some circumstances, the former for author-date.

> **reprintfrom:** This type provides a replacement for the old reprinttitle mechanism described below. As in the standard styles, it presents a fuller reference to the reprinted material than does origpubin, and is designed particularly for presenting pieces formerly printed in other collections or perhaps essays collected from various periodicals. (In biblatex-chicago it contains some kludges to cope with possible babel language environments, so if you find it behaving oddly please let me know, including whether you are using babel [which I've tested] or polyglossia [which I've tested somewhat less].)

Now, the Chicago-specific types:

> **commenton:** I designed this relatedtype to facilitate citation of online comments, though it works slightly differently in the two entry types in which it is available, online and review (with its clone suppperiodical) (14.208–10). In both types it allows you to mimic thread structure by citing a chain of replies to comments on posts, etc., all in a single entry, while also simplifying your .bib entries. This simplification works differently depending on whether the comment itself has no specific title, as always in review entries, or does have such a title, as especially in online social media

entries. In the former case, as you can see from ac:comment, the related apparatus allows you, and indeed encourages you, not to provide a title at all, as contrasted with the old system, still available of course, where your title field contained special formatting for the title of the blog on which this entry is a comment. Note also here the eventdate field, a requirement, with its optional time stamp, which helps to differentiate multiple comments by the same author posted on the same day. The options field can be used to prevent the entry appearing in the bibliography, and you can also provide a url specific to the comment, though this is by no means necessary.

In social media threads comments and replies may well have their own title, so in such a case you still need a title field, which will be followed by the related-string. In such online entries, the *only way to cite these comments* is by using the commenton relatedtype (licis:diazcomment). Note that, unlike review entries, the date, and possible time stamp, of a comment should appear in the date field, not eventdate. (Other fields, like url and options, have much the same uses as in review entries.) If, in other online entries, you decided *not* to use commenton in an entry like braun:reply, and simply use a specially-crafted titleaddon field, you lose the possibility of having two dates in the entry, one for the comment and one for the original post, though to be fair it does end up looking like the example in 14.210, where it is ambiguous to which part of the citation the date applies.

As for the thread structure, I've not tested how far down the rabbit hole you can go, but a series of entries linked one to the next by this relatedtype will all turn up if you cite the first in the chain, though of course you can use the technique merely as a convenient way to structure and simplify your .bib file, without creating chains longer than 2 entries. The default connecting string is the localized commenton, but you can use relatedstring to change it to "reply to" or whatever else you need. I've tried to follow the rules for abbreviating parts of the various works included in the one reference, though in truth the *Manual* provides no examples. Depending on whether the various parts have already been cited or not your references can take on quite varied appearances. Let me know if something looks wrong to you. Cf. ac:comment, diaz:surprise, ellis:blog, and licis:diazcomment for the use of the new relatedtype; amlen:hoot, amlen:wordplay, and viv:amlen for blogs and comments without the related mechanism.

There are a few other things to remember. As with the next two relatedtypes, biblatex-chicago will automatically set the related option to true entry-by-entry to ensure that the full data appears both in notes and in the bibliography. If your parent entry has no title of its own, then, as with the reviewof relatedtype, it will use the related functionality also in short notes, which means that if you want to provide a shorttitle for them then it goes in the *child* entry rather than the parent. Finally, the title-less comments are prime candidates for the shortextrafield option, which prints a disambiguating field after short notes when they would otherwise be indistinguishable. So endemic is this situation in this context that I've set a default means of disambiguating them, which is the date and time in online entries or the eventdate and eventtime in review and suppperiodical, though you can of course override these defaults by setting the shortextra options yourself. See their documentation in section 4.4.3, below.

maintitle: The 17th edition of the *Manual* has deployed, in at least two contexts, a notable syntactic change in the presentation of works that form part of other, larger works. Generally, the order of presentation, in biblatex terms, has always been title – booktitle – maintitle, in increasing order of generality. In the vast majority of cases this order still holds, but in TV episodes, for one example, the recommendation now is to present the name of the series (booktitle) *before* the name of the episode (title). (See the **video** type in section 4.1, above). The other context in which this reversal occurs is multi-volume works (14.116–22). Here, the preferred format, at least for notes, appears to be maintitle – [book]title or, when all three titles are present, title – maintitle – booktitle. The *Manual* doesn't carry this reordering through with absolute consistency, but I think it important at least to offer it as a

possibility to users of biblatex-chicago, hence the maintitle relatedtype, which is currently the only way to achieve this reversal in this context.

In its simplest usage, to document one volume of a multi-volume set, you would have, e.g., an mvcollection entry with relatedtype maintitle, and a related field pointing to a collection entry. When you cite the mvcollection entry itself, you'll get a long note like *MVCollTitle*, vol. 1, *CollTitle*, and a short note like *MVCollTitle*, vol. 1., or, with a postnote field, *MVCollTitle*, 1:12, as the specification requires. If you wanted to cite one essay in the collection, then you would, additionally, need an incollection entry with the maintitle relatedtype and a related field pointing to the mvcollection entry already mentioned, so you're creating a chain of three different related entries but presenting them in one reference, i.e., (long form) "InCollTitle," in *MVCollTitle*, vol. 1, *CollTitle*, and (short form) "InCollTitle." It's important to keep in mind here that, in effect, you're *not* actually citing the mvcollection entry, but the one volume of it represented by the collection entry, or indeed an essay in that one volume.

Now, for the details, which are many. First, any mv* entry without the maintitle relatedtype should behave just as it always has, and can still happily be used as the target of crossref fields to supply a maintitle to other entries. The abbreviated references created when you have several, e.g., books crossref'd to the same mvbook are still available, assuming you enable them with the booklongxref option. You can happily mix the new and the old methods of presentation in your documents, but please don't mix them within individual entries, which means that if you are using a crossref field to an mvcollection entry in a collection entry, say, and the collection entry is itself the target of the mvcollection entry's related field, please be careful not to cite that collection entry independently, as it can lead to unexpected results. (If things don't look right to you, try eliminating the use of crossref entirely from these related chains and see if that helps, then send me a bug report if it does.) This restriction also means that, although the *Manual* prefers the maintitle-first format in notes and allows either syntax in the bibliography, nonetheless with biblatex-chicago whichever syntax you choose for the notes will also appear in the bibliography.

As for automatically abbreviating references using the maintitle relatedtype, this works differently depending on whether the related chain consists of 2 or 3 works. In 2-work chains (*MVCollTitle*, *CollTitle*), it's actually the first that needs abbreviating, and this didn't look right, so these entries will always print in full. (You can still regulate how much information appears in the references to individual volumes by regulating how much information appears in the .bib entries for those volumes. In the harleymt:* entries I've used as examples below, the individual volumes have a crossref field to the multi-volume work, so they inherit the publisher and location, for example. If you were to omit the crossref field you would always get an abbreviated reference which, were it to appear *after* a reference to the whole multi-volume work, would let that reference give the complete publication details and itself behave like a normal abbreviated cross reference to it.)

In 3-work chains, when you've cited more than one "InCollTitle" from a given *CollTitle*, you can choose for the short note version of the second and third titles (with just volume number rather than full *CollTitle*) to appear in the bibliography and in long notes after the first one. This is controlled using the same booklongxref option as you would use to control the old automatic abbreviation mechanism. See under that option in section , below.

As with the other two relatedtypes I've added to biblatex's standard six, the maintitle type is somewhat restricted in its relevance. If you want to use a three-work chain to cite one part of one volume, then this is possible only by starting with the following entry types: bookinbook, inbook, incollection, inproceedings, and letter. All two-work chains must start with one of the mv* types. As might be apparent from the previous list, mvreference entries are special, in that their related field should point to an inreference entry if you want to cite an entry in an "alphabetically arranged work", or to a reference entry otherwise. In other words, mvreference entries should only ever be used in 2-work chains.

It's possible it may have occurred to you that this relatedtype could, given the presence of a many-volumed collection, require rather a lot of extra entries in your .bib files, i.e., one extra mv* entry for every volume of the collection you wish to cite. Borrowing an idea from the multivolume relatedtype, you can put the entry keys of *all* the individual volumes into a single related field in a single mv* entry, and biblatex-chicago will still allow you to cite each volume independently, and for each to appear independently in the bibliography, too, unlike the multivolume mechanism. Here's how it works. When Biber detects more than one entrykey in an mv* entry with maintitle relatedtype, it produces a series of clones of the mv* entry, each with the same relatedtype and a related field containing exactly *one* of the entrykeys from the original related field. It gives each of these clones its own entrykey, of the form mventrykey-singlevolumeentrykey, and it is these virtual, cloned entries that you should cite. Such entries don't exist in your .bib file, but you can see them in your .bbl file, assuming you've actually cited any of them. The original mventrykey, in this case, refers merely to the original mv* entry, as though it had never had a related field, so it's available for citing the multi-volume set as a whole, should that be necessary. Indeed, to make the virtual clones available to Biber (and biblatex) in the first place, you do need to cite (or \nocite) the original mv* entry somewhere in your document.

As an example of how this might look, consider the three entries from notes-test.bib: harleymt:hoc, harleymt:ancient:cart, and harleymt:cartography. The first, an mv-collection entry, has a related field containing both of the others (collection entries), and in cms-notes-sample.tex you'll see citations of harleymt:hoc, harleymt:hoc-harleymt:ancient:cart, and harleymt:hoc-harleymt:cartography, which are themselves mvcollection entries. The latter two don't exist in the .bib file, only in the .bbl file, where you'd see that each has a related field pointing to the entrykey that forms the second half of its own hyphenated key. If I hadn't somewhere cited harleymt:hoc then Biber would give up entirely because it wouldn't know where to find the two hyphenated keys.

A similar problem arises when you create a three-work chain in which the first, e.g., incollection, entry contains a related field pointing to just such a virtual, cloned entry. In this case, if you haven't already cited (or \nocite'd) the cloned entry, Biber really gets, understandably, confused. As a convenience feature for this situation, I have included a very slightly modified version of the maintitle relatedtype, called maintitlenc, the "nc" standing for \nocite. As you might have guessed, every clone produced by an mv* entry with multiple entrykeys in its related field and maintitlenc as its relatedtype will automatically be \nocite'd, and will then be available for inclusion in another entry's related field. The maintitlenc type differs in no other way whatever from the maintitle type.

In general, the maintitle relatedtype attempts to follow the Chicago specification with as little intervention needed from the user as possible. To that end, biblatex-chicago automatically sets the related option to true entry-by-entry to ensure that the full data appears both in notes and in the bibliography. It also attempts to spot duplicate authors or editors and to print them only when needed, and in its short-note version uses the volume and part information from the related collection entry, say, to specify the labeltitle which comes from the mvcollection. If you want the mv* entry's volumes data to appear in notes and bibliography, you can do so by setting the hidevolumes option to false either in the preamble or in the options field of the entry referenced by the mv* entry's related field (cf. harleymt:cartography).

Another, trickier intervention involves the problem of sorting entries in the bibliography. The *Manual's* rules are, basically, to sort by name, then title, then year, and as every mv* entry citing the same multi-volume work will basically have identical values for all three, the sorting order in the bibliography will fall back on the order in which such works are cited, which may not be what you want. If the pertinent related fields in your .bib file only contain one entrykey, then you can use a series of sortkey, sorttitle, or sortyear fields to arrange the volumes as you wish. If you are using Biber's cloning facilities, however, any such fields in the mv* entry will be copied into all of its clones, so you'll be back to square one. My current

maintitlenc

sorttitle   solution to this dilemma is to treat the sorttitle field as special, so that in mv* entries with relatedtype maintitle or maintitlenc and a related field containing more than one entrykey any sorttitle field is indeed copied to all of the clones, but it is also modified by appending the contents of the clone's related field to the end of it. (If there is no sorttitle field, then the clones will have none, either.) In effect, the alphabetical order of the entrykeys in the mv* entry's related field determines the sorting order of the clones each of which contains a related field holding one of those keys. (In the examples from notes-test.bib, harleymt:hoc retains its original sorttitle field, and sorts first, then harleymt:hoc-harleymt:ancient:cart sorts before harleymt:hoc-harleymt:cartography — the second half of the key, after the hyphen, is the relevant part, and is what appears appended to the original sorttitle field.) It is rather onerous, I know, to be required to choose entrykeys that sort properly; if I come up with something better I'll include it in a later release.

Finally, although I've tested this functionality extensively, it's new and rather complicated. If something doesn't work right please let me know at the email address at the head of this documentation.

reviewof: Philip Kime's biblatex-apa package includes this type, and user Bertold Schweitzer suggested it might be a useful addition to biblatex-chicago, so I've added it to the standard six detailed above. It differs from all of them, and resembles commenton and maintitle, in that it prints the relatedstring (\bibstring{reviewof} by default) and the data from the related entry in the middle of the parent entry, rather than at the end. It differs from commenton in that it's not possible to create a chain of such entries to mimic online thread structures. Finally, it differs from all other relatedtypes in being available only in article and review entries (along with the latter's clone, suppperiodical).

In article entries it replaces the titleaddon with the relatedstring followed by the title of the child entry, and in review entries it replaces the title with the same two components. In both types these components will optionally be followed by the author, editor, translator, etc., of the reviewed item, and then any child titleaddon may optionally appear at the end, allowing maximum flexibility when presenting, for example, reviews of live performances.

This mechanism automates both the provision of the localized \bibstring and also the formatting of the title of the reviewed work, and it also obviates the need to use any of the \partedit macros in this context. Further, biblatex-chicago automatically sets the related option to true entry-by-entry to ensure that the full data appears both in notes and in the bibliography. Finally, this relatedtype has the further peculiarity that, in review and suppperiodical entries only, it uses the related functionality also in short notes, which means that if you want to provide a shorttitle for short notes then it goes in the *child* entry rather than the parent. Please remember, too, that the standard way of presenting reviewed works is still available if the mechanism doesn't work for you in a particular context.

short: This relatedtype is like the default type, only it prints *short* references rather than long ones. There is no default relatedstring for this type, so if you leave that field blank then the short references will simply appear at the end of the long note or bibliography entry.

reprinttitle   **NB: If you have been using this feature, you may want to have a look at the** relatedtype reprintfrom, **documented above, for a better solution to this problem, one that also allows you to change the introductory string using the** relatedstring **field. The** reprinttitle **field will continue to work as before, however.** At the request of Will Small, I have included a means of providing the original publication details of an essay or a chapter that you are citing from a subsequent reprint, e.g., a *Collected Essays* volume. In such a case, at least according to the *Manual* (14.181), such details needn't be provided in notes, only in the bibliography, and then only if these details are "of specific interest." The data would follow an introductory phrase like "originally published as," making the problem strictly parallel to that of including details of a work in the original language alongside the details

of its translation. I have addressed the latter problem with the userf field, which provides a sort of cross-referencing method for this purpose, and reprinttitle works in *exactly* the same way. In the .bib entry for the reprint you include a cross-reference to the cite key of the original location using the reprinttitle field (which it may help mnemonically to think of as a "reprinted title" field). The main difference between the two forms is that userf prints all but the author of the original work, whereas reprinttitle suppresses both the author and the title of the original, giving only the more general details, beginning with, e.g., the journaltitle or booktitle and continuing from there. The string prefacing this information will be "Reprinted from." Please see the documentation on **userf** below for all the details on how to create .bib entries for presenting your data.

userf This is one of the supplemental fields which biblatex provides, and is used by biblatex-chicago for a very specific purpose. When you cite both a translation and its original, the *Manual* (14.99) recommends that, in the bibliography at least, you combine references to both texts in one entry, though the presentation in notes is pretty much up to you. In order to follow this specification, I have provided a third cross-referencing system (the others being crossref and xref), and have chosen the name userf because it might act as a mnemonic for its function.

In order to use this system, you should start by entering both the original and its translation into your .bib file, just as you normally would. The mechanism works for any entry type, and the two entries need not be of the same type. In the entry for the *translation*, you put the cite key of the original into the userf field. In the *original's* entry, you need to include something that will prevent the entry from being printed separately in the bibliography — skipbib in the options field will work, as would something in the key-words field in conjunction with a notkeyword= switch in the \printbibliography command. In this standard case, the data for the translation will be printed first, followed by the string originally published as, followed by the original, author omitted, in what amounts to the same format that the *Manual* uses for long footnotes (furet:passing:eng, furet:passing:fr). As explained above (**origlanguage**), I have also included a way to modify the string printed before the original. In the entry for the *translation*, you put the original's language in origlanguage, and instead of originally published as, you'll get French edition: or Latin edition:, etc. (aristotle:metaphy:gr, aristotle:metaphy:trans). **NB:** You can use the relatedtype origpubas to replicate the userf functionality, and you can also customize the relatedstring field to achieve the same result as with origlanguage.

4.3 Commands

In this section I shall attempt to document all those commands you may need when using biblatex-chicago-notes that I have either altered with respect to the standard provided by biblatex or that I have provided myself. Some of these, unfortunately, will make your .bib file incompatible with other biblatex styles, but I've been unable to avoid this. Any ideas for more elegant, and more compatible, solutions will be warmly welcomed.

4.3.1 Formatting Commands

These commands allow you to fine-tune the presentation of your references in both notes and bibliography. You can find many examples of their usage in notes-test.bib, and I shall try to point you toward a few such entries in what follows. **NB:** biblatex's \mkbibquote command is mandatory in some situations. See its entry below.

\autocap Version 0.8 of biblatex introduced the \autocap command, which capitalizes a word inside a note or bibliography entry if that word follows sentence-ending punctuation, and leaves it lowercase otherwise. As this command is both more powerful and more elegant than the kludge I designed for a previous version of biblatex-chicago-notes (see **\bibstring** below), you should be aware that the use of the single-letter \bibstring commands in your .bib file is obsolete.

In order somewhat to reduce the burden on users even further, I have, following biblatex's example, implemented a system which automatically tracks the capitalization of certain fields in your .bib file. I chose these fields after a non-scientific survey of entries in my

own databases, so of course if you have ideas for the extension of this facility I would be most interested to hear them. In order to take advantage of this functionality, all you need do is begin the data in the appropriate field with a lowercase letter, e.g., note = {with the assistance of X}. If the data begins with a capital letter — and this is not infrequent — that capital will always be retained. (cf., e.g., creel:house, morgenson:market.) If, on the other hand, you for some reason need such a field always to start with a lowercase letter, then you can try putting an empty set of curly braces {} at the start, which turns off the mechanism without printing anything itself. Here, then, is the complete list of fields where this functionality is active:

1. The **addendum** field in all entry types.
2. The **booktitleaddon** field in all entry types.
3. The **edition** field in all entry types. (Numerals work as you expect them to here.)
4. The **maintitleaddon** field in all entry types.
5. The **note** field in all entry types.
6. The **part** field in entry types that use it.
7. The **prenote** field prefixed to citation commands.
8. The **relatedstring** field in all entry types.
9. The **shorttitle** field in the review (suppperiodical) entry type and in the misc type, in the latter case, however, only when there is an entrysubtype defined, indicating that the work cited is from an archive.
10. The **title** field in the review (suppperiodical) entry type and in the misc type, in the latter case, however, only when there is an entrysubtype defined, indicating that the work cited is from an archive.
11. The **titleaddon** field in all entry types.
12. The **type** field in artwork, audio, image, music, suppbook, suppcollection, and video entry types.

In any other cases — and there are only two examples of this in notes-test.bib (centinel: letters, powell:email) — you'll need to provide the \autocap command yourself. Indeed, if you accidentally do so in one of the above fields, it shouldn't matter at all, and you'll still get what you want, but taking advantage of the automatic provisions should at least save some typing.

\bibstring This is a very powerful mechanism to allow biblatex automatically to provide a localized version of a string, and to determine whether that string needs capitalization, depending on where it falls in an entry. Biblatex 0.7 introduced functionality which sometimes allows you simply to input, for example, newseries instead of \bibstring{newseries}, the package auto-detecting when a bibstring is involved and doing the right thing, though in all such cases either form will work. This functionality is available in the series field of article, jurisdiction, legislation, periodical, and review entries; in the type field of audio, manual, music, patent, report, suppbook, suppcollection, thesis, and video entries; in the location field of patent entries; in the language field in all entry types; in the nameaddon field in customc entries; and in the editor[abc]type and nameatype fields in all entry types. There may be other places where biblatex's standard styles support this feature, and I shall add them when they come to my attention.

\foottextcite These two commands look like citation commands, but are in fact wrappers for cus-
\foottextcites tomizing the behavior of the \textcite and \textcites citation commands when they are used inside a foot- or an endnote. By default, in such a context these commands print the name of the author(s) followed by the *short* citation or citations, i.e., usually title only, enclosed within parentheses. You can change the way the citation part is presented by using \renewcommand in your preamble. The default definitions are: {\addspace\headlessparenshortcite} & {\addspace\headlessparenshortcites}. If you wanted to return to the default behavior of previous releases of biblatex-chicago you could change the first to: {\newcunit\bibstring{in}\addspace\headlesscite}, and the second similarly, only using \headlesscites. (There is also, by the way, a \headless-parencite(s) command if you want to retain the long citations inside the parentheses.)

\letterdatelong I have provided this macro mainly for use in the optional postnote field of the various citation commands. When citing a letter (published or unpublished, letter or misc), it

may be useful to append the date to the usual short note form in order to disambiguate references. This macro simply prints the date of a letter, or indeed of any other sort of correspondence, in day-month-year order, as recommended by the *Manual* (14.224). (If your main document language isn't American, it's better just to use the standard biblatex command \printorigdate.)

\mkbibquote This is the standard biblatex command, which requires attention here because it is a crucial part of the mechanism for the "American" punctuation system. If you look in chicago-notes.cbx you'll see that the quoted fields, e.g., an article or incollection title, have this command in their formatting, which does most of the work for you. If, however, you need to provide additional quotation marks in a field — a quoted title within a title, for example — then you may need to use this command so that any following period or comma will be brought within the closing quotation marks. Its use is *required* when the quoted material comes at the end of a field, and I recommend always using it in your .bib database, as it does no harm even when that condition is not fulfilled. A few examples from notes-test.bib should help to clarify this.

In an article entry, the title contains a quoted phrase:

```
title = {Diethylstilbestrol and Media Coverage of the
  \mkbibquote{Morning After} Pill}
```

Here, because the quoted text doesn't come at the end of title, and no punctuation will ever need to be drawn within the closing quotation mark, you could instead use \enquote{Morning After} or even `Morning After'. (Note the single quotation marks here — the other two methods have the virtue of taking care of nesting for you.) All of these will produce the formatted "Diethylstilbestrol and Media Coverage of the 'Morning After' Pill." Here, by contrast, is a book title:

```
title = {Annotations to \mkbibquote{Finnegans Wake}}
```

Because the quoted title within the title comes at the end of the field, and because this bibliographical unit will be separated from what follows by a period in the bibliography, then the \mkbibquote command is necessary to bring that period within the final quotation marks, like so: *Annotations to "Finnegans Wake."*

Let me also add that this command interacts well with Lehman's csquotes package, which I highly recommend, though the latter isn't strictly necessary in texts using an American style, to which biblatex defaults when csquotes isn't loaded.

\postvolpunct The *Manual* (14.116) unequivocally prescribes that when a volume number appears immediately before a page number, "the abbreviation *vol.* is omitted and a colon separates the volume number from the page number with no intervening space." The treatment is basically the same whether the citation is of a book or of a periodical, and it appears to be a surprising and unwelcome feature for many users, conflicting as it may do with established typographic traditions in a number of contexts. Clea F. Rees requested a way to customize this, so I have provided the \postvolpunct command, which prints the punctuation between a volume number and a page number. It is set to \addcolon by default, except when the current language of the entry is French, in which case it defaults to \addcolon\addspace. You can use \renewcommand{\postvolpunct}{...} in your preamble to redefine it, but please note that the command only applies in this limited context, not more generally to the punctuation that appears between, e.g., a volume and a part field.

\reprint This and the following 7 macros all help biblatex-chicago-notes cope with the fact that many bibstrings in the Chicago system differ between notes and bibliography, the former sometimes using abbreviated forms when the latter prints them in full. In the current case, if a book is a reprint, then the macro \reprint, followed by a comma, could go in the location field before the city of publication. Simply putting "reprint" into the pubstate field is a simpler way to achieve the same result (aristotle:metaphy:gr, schweitzer:bach). See location and pubstate in section 4.2, above.

NB: The rules for employing abbreviated or full bibstrings in the *Manual* are remarkably complex, but I have attempted to make them as transparent for users as possible. In biblatex-chicago-notes, if you don't see it mentioned in this section, then in theory you should always provide an abbreviated version, using the `\bibstring` mechanism, if necessary (babb:peru). The standard biblatex bibstrings should also work (palmatary:pottery), and any that won't should be covered by the series of macros beginning here with `\reprint` and ending below with `\parttransandcomp`.

\partcomp Since the *Manual* specifies that the strings editor, translator, and compiler all require different forms in notes and bibliography, and since it mentions these three apart from all the others biblatex provides (annotator, commentator, et al.), and further since it may indeed happen that the available fields (editor, namea, translator, nameb, and namec) aren't adequate for presenting some entries, I have provided 7 macros to allow you to print the correct strings for these functions in both notes and bibliography. Their names all begin with `\part`, as originally I intended them for use when a particular name applied only to a specific title, rather than to a maintitle or booktitle (cf. **namea** and **nameb**, above).

In the present instance, you can use `\partcomp` to identify a compiler when namec won't do, e.g., in a note field or the like. In such a case, biblatex-chicago-notes will print the appropriate string in your references.

\partedit Use this macro when identifying an editor whose name doesn't conveniently fit into the usual fields (editor or namea). (N.B.: If you are writing in French then you no longer need to add either de or d' after this command in your .bib files. The new version of the command should take care of this automatically for you.) See chaucer:liferecords.

\partedit-andcomp As before, but for use when an editor is also a compiler.

\partedit-andtrans As before, but for when when an editor is also a translator (ratliff:review).

\partedit-transandcomp As before, but for when an editor is also a translator and a compiler.

\parttrans-andcomp As before, but for when a translator is also a compiler.

\parttrans As before, but for use when identifying a translator whose name doesn't conveniently fit into the usual fields (translator and nameb).

\suppress-bibfield[]{} Unlike the other commands presented here, this should be used in your document preamble rather than in your bibliographical apparatus. Also unlike them, it has two arguments, the first of which is optional, the second required. Jan David Hauck suggested that, in addition to the field-exclusion package options provided by biblatex-chicago (see section 4.4.2), I might also provide a general-purpose macro to clear fields from selected entry types when the package options aren't quite right for a user's particular needs. The `\suppressbibfield` command does this, so that `\suppressbibfield{note}` clears the note field from *all* entries, while `\suppressbibfield[report]{note}` clears it only from report entries. Both arguments take comma-separated lists, so to suppress titleaddon and volumes fields from report and manual entries, your preamble could contain `\suppressbib-field[report,manual]{titleaddon,volumes}`.

A few usage notes are in order. First, you can use as many calls to the command in your preamble as you wish. Second, the command is a very basic user interface to biblatex's source mapping functionality (biblatex.pdf § 4.5.3), so what it does is modify what biber takes from your .bib file in order to produce the .bbl file that biblatex actually reads. As far as biblatex is concerned, the fields simply aren't there in the data source, so they can't appear anywhere in the bibliographical apparatus, whether in notes, bibliography or shorthand lists. Third, because source mapping is involved, you'll need a complete cycle of LaTeX-biber-LaTeX runs to make the commands take effect. Fourth, source mapping occurs at a very early stage in biber's operation, so if your field names or entry types are standard aliases, the command will only work on the names as they appear in your .bib file,

not as they are aliased in the .bbl file. If you have a techreport entry, for example, it won't be affected by a command that alters report entries, and a date field won't be affected by a command that suppresses the year. Fifth, the code for the command resides in biblatex-chicago.sty, so if you load the styles without loading that package it won't be available to you. Sixth and finally, the \suppressbibfield command is new and relatively untested, so please report any untoward behavior to me.

4.3.2 Citation Commands

The biblatex package is particularly rich in citation commands, some of which (e.g., \supercite(s), \citeyear) provide functionality that isn't really needed by the Chicago notes and bibliography style offered here. If you are getting unexpected behavior when using them please have a look in your .log file — there may be warnings there that alert you to undefined citation commands. Other biblatex-provided commands, though I haven't tested them extensively, should pretty much work out of the box. What remains are the commands I have found most useful and necessary for following the *Manual*'s specifications, and I document in this section any alterations I have made to these. As always, if there are standard commands that don't work for you, or new commands that would be useful, please let me know, and it should be possible to fix or add them.

A number of users have run into a problem that appears when they've used a command like \cite inside a \footnote macro. In this situation, the automatic capitalization routines will not be in operation at the start of the footnote, so instead of "Ibid.," for example, you'll see "ibid." If you need to use the \cite command within a \footnote command, the solution is to use \Cite instead. Alternatively, don't use a \footnote macro at all, rather try \footcite or \autocite with the optional prenote and postnote arguments. Cf. \Citetitle below, and also section 3.8 of biblatex.pdf.

\autocite I haven't adapted this in the slightest, but I thought it worth pointing out that biblatex-chicago-notes sets this command to use \footcite as the default option. It is, in my experience, much the most common citation command you will use, and also works fine in its multicite form, **\autocites**.

\cite* While the \cite command works just as you would expect it to, I have also provided a starred version for the rare situations when you might need to turn off the ibidem tracking mechanism. Biblatex provides very sophisticated algorithms for this mechanism, so in general you won't find a need for this command, but in case you'd prefer a longer citation where you might automatically find the shortest one, I've provided this. Of course, you'll need to put it inside a \footnote command manually. (See also section 4.4.3, below.)

\citeauthor I have adapted this standard biblatex command only very slightly to bring it into line with biblatex-chicago's needs. Its main usage will probably be for references to works from classical antiquity, when an author's name (abbreviated or not) sometimes suffices in the absence of a title, e.g., Thuc. 2.40.2–3 (14.244). You'll need to put it inside a \footnote command manually. (Cf. also entrysubtype in section 4.2, above, and please note that the new notite option (section 4.4.3) produces the same effect without the need to worry about citation commands.)

\citeincite Although the officially-sanctioned, and safest, way to present cross-references to other works is by using the related mechanism, it would appear, judging from various bug reports I've received, that there are plenty of users who need to present such references in ways that aren't supported by the related code as it stands. This new citation command is designed to allow you to present short references to other works inside other references, and to avoid some, I hope most, of the bugs associated with using cite commands in the fields of other entries. If you want to present long citations, I'm assuming that you wouldn't want to do that in fields right in the middle of the parent entry, and that, therefore, the usual methods detailed in section 4.2.1 (above) or the \fullciteincite command (below) will serve your needs. Similarly, if you want to provide short citations at the end of your parent entry, the new relatedtype short should work.

If, however, you must have a short reference in the midst of another field, then \citeincite can help. It is certified to work properly only in the addendum, annotation, annote, note, and titleaddon fields, which I hope is a broad enough choice to cover most needs. The multicite command \citeincites is also available. In previous releases of biblatex there were differences in how the case-changing backends behaved with respect to citation commands inside entry fields, but in my testing the two backends appear currently to be equivalent on this score. Still, should you run into problems using \citeincite, it might still be worth setting casechanger=latex2e in your preamble to see if that helps.

\citejournal   This command provides an alternative short form when citing journal articles, giving the journaltitle and volume number instead of the article title after the author's name. The *Manual* suggests that this format might be helpful "in the absence of a full bibliography" (14.185). It may also prove useful when you want to provide parenthetical references to newspaper articles within the text rather than in the bibliography, a style endorsed by the *Manual* (14.198). In such a case, an article's author, if there is one, could form part of the running text. As usual with these general citation commands, if you want the reference to appear in a footnote you need to put it inside a \footnote command manually.

\Citetitle   This simply prepends \bibsentence to the usual \citetitle command. Some titles may need this for the automatic contextual capitalization facility to work correctly. (Included as standard from biblatex 0.8d.)

\citetitles   Joseph Reagle noticed that, because of the way biblatex-chicago-notes formats titles in quotation marks, using the \citetitle command will often get you punctuation you don't want, especially when presenting a list of titles. I've included this multicite command to enable you to present such a list, if the need arises. Remember that you'll have to put it inside a \footnote command manually.

\footfullcite   Another standard biblatex command, modified to work properly with biblatex-chicago-notes, and provided in case you find yourself in a situation where you really need the full citation in a footnote, but where \autocite would print a short note or even engage the ibidem mechanism. This may be particularly useful if you've chosen to use all short notes by setting the short option in the arguments to \usepackage{biblatex}, yet still feel the need for the occasional full citation.

\fullcite   This, too, is a standard command, and it too provides a full citation, but unlike the previous command it doesn't automatically place it in a footnote. It may be useful within long textual notes.

\fullciteincite   Ordinarily, you can use one of the methods discussed in section 4.2.1 to present, in notes or bibliography, full references to a work related in some way to the current entry. The new command \citeincite (above) allows you to present short citations to other works within selected fields of the parent entry, while attempting to minimize the problems caused by citations within other citations. The \fullciteincite command does the same for full citations, but really is designed to work only in annotation fields, the idea being that you might want to refer in an annotated bibliography to works that you haven't cited in the main body of your text. The multicite command \fullciteincites is also available.

\gentextcite   Arne Skjærholt requested, for the author-date styles, a variant of the \textcite command that presented the author's name in the genitive case in running text, thereby simplifying certain syntactic constructions. As a convenience for users, I've also ported this to the notes & bibliography style. In most respects it behaves exactly like \textcite, on which see below. The difference is that I've added a new optional field to the front of the command to allow you to choose which declensional ending to add to the name. If you don't specify this field, you'll get the standard English "'s". If you want something different, then you'll need to present a third option to the command, like so: \gentextcite[<ending>][][]{entry:key}. You must include the two further sets of square brackets, because with only one set it will, as with other citation commands, be interpreted as a postnote, and with two a prenote and a postnote. There is a \gentextcites

command as well, and for it you may need to specify \gentextcites[<ending>]()()[][]{entry:key1}{entry:key2}, though if you don't have a pre- or postnote to the first citation you can make do with \gentextcites[<ending>](){entry:key1}{entry:key2}.

The syntax of multiple authors' names in running text is unpredictable. There is currently no way to add the genitival ending to all the names attached to a single citation key, so it will only appear at the end of a group of names in such a case. (This is in keeping with the usual syntax when referring to a multi-author work, at least in English.) When using \gentextcites, however, you can control whether the ending appears after the name(s) attached to each citation key, or whether it only appears after the names attached to the last key. By default, it only appears after the last, but the genallnames preamble and/or entry option set to true will attach the ending to each key's name(s).

\headlesscite
\headlesscites
\Headlesscite
Matthew Lundin requested a more generalized \headlesscite macro, suppressing the author's name in specific contexts while allowing users not to worry about whether a particular citation needs the long or short form, a responsibility thereby handed over to biblatex's tracking mechanisms. These citation commands attempt to fulfill this request. The (new) capitalized command, as usual, will ensure capitalization of, e.g., "ibid." at the beginning of notes, and was made necessary by fixes to a bug identified by David Purton. Please note that, in the short form, the result will be rather like a \citetitle command, which may or may not be what you want. Note, also, that as I have provided only the most flexible form of the command, you'll have to wrap it in a \footnote yourself. Please see the next entry for further discussion of some of the needs this command might help address.

\headless-
fullcite
\headless-
fullcites
I have provided these commands in case you want to print a full citation without the author's name. The *Manual* (14.78, 14.104) suggests this for brevity's sake in cases where that name is already obvious enough from the title, and where repetition might seem awkward (creel:house, feydeau:farces, meredith:letters, and sewall:letter). Letter entries and standard entries (where the author is usually the publishing organization) — and only these entries — do this for you automatically, and of course the repetition is tolerated in bibliographies for the sake of alphabetization, but in notes for other entry types this command may help achieve greater elegance, even if it isn't strictly necessary. As I've provided only the most flexible form of the command, you'll have to wrap it in a \footnote yourself.

\shortcite
I have provided this command in case, for any reason, you specifically require the short form of a note, and biblatex thinks you want something else. Again, I've provided only the most flexible form of the command, so you'll have to wrap it in a \footnote manually.

\shortcite*
The starred version of the command turns off page and citation tracking for a short note, designed particularly to prevent a noteref back reference from appearing, should you and the code have a disagreement over just when such back references might be necessary.

\shorthandcite
At the request of Kenneth Pearce, I have included this command which always prints the shorthand, even at the first citation of a given work. Again, I've only provided the most flexible form of the command, so you'll need to place it inside parentheses or wrap it in a \footnote manually.

\shorthandcite*
The starred version of the command turns off page and citation tracking for a shorthand note, designed particularly to prevent a noteref back reference from appearing, should you and the code have a disagreement over just when such back references might be necessary.

\shorthand-
refcite
This command, like the next, forces the printing of a back reference when you are using the new package option noteref, only it prints a shorthand note rather than a short note. It's the opposite of \shorthandcite*.

\shortrefcite
The new package option noteref provides for the printing of back references from short notes to their corresponding long notes. Biblatex-chicago provides several options to help

you control when such back references should appear, but as there may well be occasions when you and the code disagree, this command forces the printing of a short note with a back reference. It's the opposite, then, of \shortcite*, which prevents such a back reference from appearing.

\surnamecite This command is analogous to \headlesscite, but whereas the latter allows you to omit an author's name when that name is obvious from the title of a work, \surnamecite allows you to shorten a full note citation in contexts where the full name(s) of the author have already been provided in the text. In short notes this falls back to the standard format, but in long notes it simply omits the given names of the author and provides only the surname, along with the full data of the entry (cf. 14.57).

\textcite Norman Gray started a discussion on Stackexchange which established both that biblatex had begun including a \textcite command in its verbose styles and that biblatex-chicago-notes hadn't kept up. In that thread Audrey Boruvka provided some code, adapted from verbose.cbx, to provide such a command for the Chicago notes & bibliography style. More recently, Rasmus Pank Rouland pointed out some changes in biblatex that made the \textcites command fit more elegantly into the flow of text. I've adapted this solution in this release. I'm still not entirely certain how best to accommodate this request within the package, but there are now at least commands (\textcite and \textcites) for users to test. Their functionality is a little complicated. In the main text, they will provide an author's name(s), followed immediately by a foot- or endnote which contains the full (or short) reference, following the usual rules. If you use \textcite inside a foot- or endnote, then the default behavior, for both \textcite and \textcites, specifies that you'll get the author's name(s) followed by a headless *short* citation (or citations) placed within parentheses. Such parentheses are generally discouraged by the *Manual* (14.38), but are nonetheless somewhat better than other solutions for smoothing the syntax of sentences that include such a citation. I have made the citation short, i.e., title only, because this again seems likely to be the least awkward solution syntactically. If you want to configure this behavior for either citation command, please see \foottextcite and \foottextcites in section 4.3.1.

If you look at chicago-notes.cbx, you'll see a number of other citation commands, but those are intended for internal use only, mainly in cross-references of various sorts. Use at your own risk.

4.4 Package Options

4.4.1 Pre-Set biblatex Options

Although a quick glance through biblatex-chicago.sty will tell you which biblatex options the package sets for you, I thought I might gather them here also for your perusal. These settings are, I believe, consistent with the specification, but you can alter them in the options to biblatex-chicago in your preamble or by loading the package via \usepackage [style=chicago-notes]{biblatex}, which gives you the biblatex defaults unless you redefine them yourself inside the square brackets.

abbreviate= By default, biblatex-chicago-notes prints the longer bibstrings, mainly for use in the bibfalse liography, but since notes require the shorter forms of many of them, I've had to define many new strings for use there.

autocite= Biblatex-chicago-notes places references in footnotes by default.
footnote

citetracker= The citetracker for the \ifciteseen test is enabled globally.
true

alldates=comp The specification calls for the long format when presenting dates, slightly shortened when presenting date ranges.

alltimes=12h In entries which print time stamps, they will, when the stamp is part of a date, eventdate, or origdate, appear in 12-hour format, i.e., "4:45 p.m." Stamps that are part of a urldate are, by default, controlled by the urltime option, which is set to 24h. See that option below, and table 3.

dateabbrev=false	The *Manual* prefers to use full month names in the notes & bibliography style.
datecirca=true	This option enables biblatex's enhanced "circa" date specification, which given a date like 1989~ will print [ca. 1989]. Cf. table 3.
dateuncertain=true	This option enables biblatex's enhanced "uncertain" date specification, which given a date like 1989? will print [1989?]. A field like 1989% is both "circa" *and* "uncertain," like so: [ca. 1989?]. Cf. table 3.
datezeros=false	This ensures that leading zeros don't appear in date specifications.
ibidtracker=constrict	This enables the use of the ibidem mechanism in notes, but only in the most strictly-defined circumstances. Whenever there might be any ambiguity, biblatex should default to printing a more informative reference. Remember also that you can use the \cite* command to disable this functionality in any given reference, or indeed one of the fullcite commands if you need the long note form for any reason. Please see section 4.4.3 for options (noibid and useibid) managing how biblatex-chicago presents ibidem references, as the defaults have changed in the 17th edition (14.34).
idemtracker=false	Roger Hart suggested that it might be helpful, despite the *Manual's* objections (14.35), to be able to turn on biblatex's idemtracker. This replaces, in notes only, authors' names with the string "Idem" when a work by the same author follows a different work by that author, two consecutive references to the same work by the same author generating, of course, "Ibid." Indeed, if you are going to use the idemtracker, you should also set the package option useibid to true, so that you don't get a mix of "Idem" and the new 17th-edition ibidem behavior which doesn't print "Ibid." You can turn this all on when loading biblatex-chicago by setting idemtracker=constrict,useibid=true. It works very much like the standard biblatex styles which include this option, so that you never get "Idem" in long notes, but only in short ones, and (ideally) never when the repeated name might be somewhat ambiguous. Also, if you wish the localized string to be appropriately gendered, you need to employ the biblatex field gender, on which see biblatex.pdf § 2.3.3.
labeltitle=true	This option enables biblatex-chicago-notes to disambiguate shortened citations to different sources which might otherwise be confusingly identical. Though I've set it in biblatex-chicago.sty, you'll need to set the shortextrafield option yourself in order for it to have any visible effect. See the documentation of that option in section 4.4.3, below.
loccittracker=constrict	This allows the package to determine whether two consecutive citations of the same source also cite the same page of that source. In such a case, Ibid alone will be printed, without the page reference, following the specification (14.29).
maxbibnames=10 minbibnames=7	These two options control the number of names printed in the bibliography when that number exceeds 10. These numbers follow the recommendations of the *Manual* (14.76), and they are different from those for use in notes. With biblatex 1.6 you can no longer redefine maxnames and minnames in the \printbibliography command at the bottom of your document, so biblatex-chicago now does this automatically for you, though of course you can change them in your document preamble.
pagetracker=true	This enables page tracking for the \iffirstonpage and \ifsamepage commands for controlling, among other things, the printing of "Ibid." It tracks individual pages if LaTeX is in oneside mode, or whole spreads in twoside mode.
related=bib	This is the standard biblatex bibliography option, but I have given it some extra settings and also added entry and type options as well. By default it enables the use of related functionality in the bibliography only, not in long notes. You can set it either in the preamble or in individual entries to enable the functionality in long notes only (notes), in both notes and bibliography (true), or in neither (false). When you use the commenton, maintitle, or reviewof relatedtypes, biblatex-chicago automatically sets this option to true on an entry-by-entry basis, as these relatedtypes require this for proper functioning. Cf. coolidge:speech, weed:flatiron.
sortcase=false	This turns off the sorting of uppercase and lowercase letters separately, a practice which the *Manual* doesn't appear to recommend.

sorting= \cms@choose	This new setting tests whether you are using Biber as your backend, and if so enables a custom biblatex-chicago sorting scheme for the bibliography (cms). If you are using any other backend, it reverts to the biblatex default (nty). Please see the discussion of \DeclareSortingTemplate just below.
timezones=true	If you provide a timezone for a time stamp, usually using one of the timezone fields, this option ensures it will be printed.
uniquename= minfull	This enables the package to distinguish, in short notes, different authors who share a surname, using initials in the first instance, and whole names if initials aren't enough (14.32).
urltime=24h	In entries with urldate fields containing time stamps, that stamp will by default appear in 24-hour format, i.e., "16:45." Cf. alltimes, above, urlstamp in section 4.4.2 below, and table 3.
[standard] useeditor=false usenamec=false	In standard entries any editors' or compilers' names appear after the title, according to 14.259, so these entry-type-specific options encode this. You can, of course, override these defaults in your preamble, should you deem it necessary.
usetranslator =true	This enables automatic use of the translator at the head of entries in the absence of an author or an editor. In the bibliography, the entry will be alphabetized by the translator's surname. You can disable this functionality on a per-entry basis by setting usetranslator=false in the options field. Cf. silver:gawain.

Other biblatex Formatting Options

I've chosen defaults for many of the general formatting commands provided by biblatex, including the vertical space between bibliography items and between items in the list of shorthands (\bibitemsep and \lositemsep). I define many of these in biblatex-chicago.sty, and of course you may want to redefine them to your own needs and tastes. It may be as well you know that the *Manual* does state a preference for two of the formatting options I've implemented by default: the 3-em dash as a replacement for repeated names in the bibliography (14.67–71, and just below); and the formatting of note numbers, both in the main text and at the bottom of the page / end of the essay (superscript in the text, in-line in the notes; 14.24). The code for this last formatting is also in biblatex-chicago.sty, and I've wrapped it in a test that disables it if you are using the memoir class, which I believe has its own commands for defining these parameters. You can also disable it by using the footmarkoff package option, on which see below.

Gildas Hamel pointed out that my default definition, in biblatex-chicago.sty, of biblatex's \bibnamedash didn't work well with many fonts, leaving a line of three dashes separated by gaps. He suggested an alternative, which I've adopted, with a minor tweak to make the dash thicker, though you can toy with all the parameters to find what looks right with your chosen font. The default definition is:

`\renewcommand*{bibnamedash}{\rule[.4ex]{3em}{.6pt}}.`

losnotes & losendnotes	At the request of Kenneth Pearce, I have added two bibenvironments to chicago-notes .bbx, for use with the env option to the \printshorthands command. The first, losnotes, is designed to allow a list of shorthands to appear inside footnotes, while losendnotes does the same for endnotes. Their main effect is to change the font size, and in the latter case to clear up some spurious punctuation and white space that I see on my system when using endnotes. (You'll probably also want to use the option heading=none in order to get rid of the [oversized] default, providing your own within the \footnote command.) If you use a command like \printbiblist{shortjournal} to print a list of journal abbreviations, you can use the sjnotes and sjendnotes bibenvironments in exactly the same way. Please see the documentation of shorthand and shortjournal in section 4.2 above for further options available to you for presenting and formatting these two types of biblist.
\Declare- Labelname	Formerly available only to those using Biber, but since version 3.0 handled by biblatex itself, \DeclareLabelname allows you to add name fields for consideration when biblatex is attempting to find a shortened name for short notes. This, for example, allows a compiler (=namec) to appear at the head of short notes without any other intervention from

the user, rather than requiring a shortauthor field as previous releases of biblatex-chicago did. The default search order for the Chicago styles is shortauthor, author, shorteditor, namea, editor, nameb, translator, namec. You can set the option use<name>=false in entries or when loading biblatex-chicago to exclude individual fields from appearing in short notes, or indeed at the head of long notes and bibliography entries. See the documentation of those name types in section 4.2 for further details.

\Declare-
Sorting-
Template

I have provided, using this declaration, a custom sorting algorithm for the bibliography. The idea is that biblatex can use any field whatsoever for sorting, so that a great many more entries will be sorted correctly automatically rather than requiring manual intervention in the form of a sortkey field or the like. Code in biblatex-chicago.sty loads the custom scheme "cms," a Chicago-specific variant of the default nty. (You can find its definition in chicago-notes.cbx.) The advantages of this scheme are, specifically, that any entry headed by one of the supplemental name fields (name[a-c]), a manual or a standard entry headed by an organization, or an article or review entry headed by a journaltitle won't need a sortkey set. Further, the use<name>=false options will remove any name field from the sorting order, again reducing the need for user intervention.

4.4.2 Pre-Set chicago Options

bookpages=
true

At the request of Scot Becker, I have included this rather specialized option, which controls the printing of the pages field in book entries. Some bibliographic managers, apparently, place the total page count in that field by default, and this option allows you to stop the printing of this information in notes and bibliography. It defaults to true, which means the field is printed, but it can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field (though rather than use this latter method it would make sense to eliminate the pages field from the affected entries).

doi=true

This option controls whether any doi fields present in the .bib file will be printed in notes and bibliography. At the request of Daniel Possenriede, and keeping in mind the *Manual's* preference for this field instead of a url (14.6), I have added a third switch, only, which prints the doi if it is present and the url only if there is no doi. Ryo Furue more recently requested a way to suppress the urldate when using only the doi, so I've added the onlynd switch to do this. The package default remains the same, however — it defaults to true, which will print both doi and url if both are present. The option can be set to only, onlynd, or to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field. In online entries, the doi field will always be printed, but the only switch will still eliminate any url, and onlynd will still eliminate both the url and the urldate.

eprint=true

This option controls whether any eprint fields present in the .bib file will be printed in notes and bibliography. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis, in the options field. In online entries, the eprint field will always be printed.

isbn=true

This option controls whether any isan, isbn, ismn, isrn, issn, and iswc fields present in the .bib file will be printed in notes and bibliography. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis, in the options field.

numbermonth
=true

Once again at the request of Scot Becker, I have included this option, which controls the printing of the month field in all the periodical-type entries when a number field is also present. Some bibliographic software, apparently, always includes the month of publication even when a number is present. When all this information is available the *Manual* (14.180, 14.185) prints everything, so this option defaults to true, which means the field is printed, but it can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field.

url=true

This option controls whether any url fields present in the .bib file will be printed in notes and bibliography. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis, in the options field.

Please note that, as in standard biblatex, the url field is always printed in online entries, regardless of the state of this option.

urlstamp=true This option controls whether any urltime fields, included as part of the urldate, will be printed in notes and bibliography. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field. Please note that, unlike the url option, this option *does* control what is printed in online entries.

includeall= true This is the one option that rules the seven preceding, either printing all the fields under consideration — the default — or excluding all of them. It is set to true in chicago-notes.cbx, but you can change it either in the preamble for the whole document or for specific entry types, or in the options field of individual entries. The seven individual options above are similarly available in the same places, for finer-grained control. The rationale for all of these options is the availability of bibliographic managers that help-fully present as much data as possible, in every entry, some of which may not be felt to be entirely necessary. Setting includeall to true probably works just fine for those compiling their .bib databases by hand, but others may find that some automatic prun-ing helps clear things up, at least to a first approximation. Some per-type or per-entry work afterward may then polish up the details. If you find that you need control over fields that aren't included among these options, I have provided the \suppressbibfield command for your preamble, as suggested by Jan David Hauck. It is in fact a user inter-face to the source mapping feature of biblatex, and it is something of a nuclear option, preventing fields from even appearing in the .bbl file generated by biber from your .bib database. See the \suppressbibfield command in section 4.3.1 and the source mapping docs in biblatex.pdf § 4.5.3.

addendum= true At the request of Roger Hart, I have included this option, which controls the printing of the addendum field, but *only* in long notes. It defaults to true, and can be set to false either in the preamble, for the whole document, or on a per-entry basis, in the options field.

bookseries= true According to the *Manual* (14.123), the series field in book-like entries "may be omitted to save space (especially in a footnote)." This option allows you to control the printing of that field in long notes. It defaults to true, and can be set to false either in the preamble, for the whole document, or on a per-entry basis, in the options field. Several entry types don't use this field, so the option will have no effect in them, and it is also ignored in article, misc, music, periodical, and review entries.

notefield= true As with the previous two options, Roger Hart requested an option to control the print-ing of the note field in long notes. It defaults to true, and can be set to false either in the preamble, for the whole document, or on a per-entry basis, in the options field. The option will be ignored in article, misc, periodical, and review fields.

completenotes= true This is the one option that rules the three preceding, either printing all the fields under consideration — the default — or excluding all of them from long notes. It is set to true in chicago-notes.cbx, but you can change it either in the preamble for the whole document or, for specific fields, in the options field of individual entries.

bibannotesep =vpar citeannotesep =period These options define the relation of the annotation field to the main entry, bibannotesep doing so in the bibliography and citeannotesep in long notes. (The annotation option in section 4.4.3 determines where, if anywhere, the field will appear.) Both options have the same set of keys, though they have different default settings if you don't define them yourself. The possible values are:

none = no punctuation at all.
space = \addspace
comma = \addcomma\addspace
period = \addperiod\addspace
colon = \addcolon\addspace
semicolon = \addsemicolon\addspace
par: This starts a new paragraph on the next line. Page breaking is strongly inhib-ited before the annotation.

vpar: This starts a new paragraph, and also inserts some vertical space before it. In long notes that vertical space is minimal (1 pt), while in the bibliography it creates a blank line. Page breaking is strongly inhibited before the annotation.

parbreak: This is the same as par, but it allows a page break to occur between the main entry and the start of the annotation.

vparbreak: This is the same as vpar, but it also allows a page break between the main entry and the annotation.

Please note that both of these options are available in the preamble both globally and per-type, and also in the options field of individual entries. Each defines a command (\bibannotesep and \citeannotesep) which appears in the annotation field's formatting directive, so it's possible to redefine these commands in your preamble if you have needs that the available values don't address. (You can also try sending an email to encourage me to add other keys.) Please also keep in mind that bibannotesep interacts with the entrybreak and formatbib options in section 4.4.3, below, to determine the general layout of the bibliography. Depending on the settings of those options, changing the bibannotesep from entry to entry may not work out well.

booklongxref= true — At the request of Bertold Schweitzer, I have included two options for controlling whether and where biblatex-chicago will print abbreviated references when you cite more than one part of a given collection or series. This option controls whether multiple book, book-inbook, collection, and proceedings entries which are part of the same collection will appear in this space-saving format. The parent collection itself will usually be presented in, e.g., a book, bookinbook, mvbook, mvcollection, or mvproceedings entry, and using crossref or xref in the child entries will allow such presentation depending on the value of the option:

true: This is the default. If you use crossref or xref fields in these entry types, by default you will *not* get any abbreviated references, either in notes or bibliography.

false: You'll get abbreviated references in these entry types both in notes and in the bibliography.

notes: The abbreviated references will not appear in notes, but only in the bibliography.

bib: The abbreviated references will not appear in the bibliography, but only in notes.

This option can be set either in the preamble or in the options field of individual entries. For controlling the behavior of inbook, incollection, inproceedings, and letter entries, please see longcrossref, below, and also the documentation of crossref in section 4.2.

compressyears =true — The *Manual* has long recommended (9.64, 14.117, 14.144), as a space-saving measure, the compression of year ranges when presenting dates. I have, finally, implemented this in the current release, and have made it the default, which you can change in your document preamble. Please note that the rules for compressing years are different from those for compressing other numbers (e.g., page numbers), and also that the compression code is in biblatex-chicago.sty, which will have to be loaded for this option to make any difference. Cf. table 3.

ctitleaddon= comma ptitleaddon= period — Roger Hart requested a way to control the punctuation printed before the titleaddon, booktitleaddon, and maintitleaddon fields. By default, this is \addcomma\addspace (ctitleaddon) for all occurrences in notes and for nearly all book- and maintitleaddons in the bibliography, while \addperiod\addspace (ptitleaddon) is the default before most titleaddons in the bibliography. If the punctuation printed isn't correct for your needs, you can set the relevant option either in the preamble or in individual entries. (Cf. coolidge: speech and schubert:muellerin.) The accepted option keys are:

none = no punctuation at all
space = \addspace
comma = \addcomma\addspace
period = \addperiod\addspace
colon = \addcolon\addspace
semicolon = \addsemicolon\addspace

If you need something a little more exotic, you can directly \renewcommand either \ctitleaddonpunct or \ptitleaddonpunct (or both) in your preamble, but it's worth remembering that the redefinition will hold for all instances, unless you use the options field in your other entries with a titleaddon field. A simpler solution might be to set the relevant option to none in your entry and then include the punctuation in the titleaddon field itself.

dashed=true Constanza Cordoni has requested a way to turn off the 3-em dash for replacing repeated names in the bibliography, and the *Manual* admits that some publishers prefer this, as the dash can carry with it certain inconveniences, especially for electronic formats (14.67). Some of biblatex's standard styles have a dashed option, so for compatibility purposes I've provided the same. By default I have set it to print the name dash, but you can set dashed=false globally, per type, or per entry to repeat names as and when required.

hidevolumes= true If both a volume and a volumes field are present, as may occur particularly in cross-referenced entries, then biblatex-chicago will ordinarily suppress the volumes field. In some instances, when a maintitle is present, this may not be the desired result. In this latter case, if the volume appears before the maintitle, this new option, set to true by default, controls whether to print the volumes field after that title or not. Set it to false either in the preamble or in the options field of your entry to have it appear after the maintitle.

jtitleaddon= space I have added the standard biblatex journaltitleaddon field to the article and review entry types, and also the titleaddon field to the periodical type, fields that may, for example, be particularly useful when you want to provide the original form of a translated journal title. The jtitleaddon option controls the separator between the main title and the addon, as with the ctitleaddon and ptitleaddon options, above, and like them is settable globally, per type, or per entry. The possible settings are the same as for those options, but the default is a space. You can redefine \jtitleaddonpunct directly if you have more unusual needs.

longcrossref= false This is the second option, requested by Bertold Schweitzer, for controlling whether and where biblatex-chicago will print abbreviated references when you cite more than one part of a given collection or series. It controls the settings for the entry types more-or-less authorized by the *Manual*, i.e., inbook, incollection, inproceedings, letter, and review. The mechanism itself is enabled by multiple crossref or xref references to the same parent, whether that be, e.g., a collection, an mvcollection, a proceedings, or an mvproceedings entry. Given these multiple cross references, the presentation in the reference apparatus will be governed by the following options:

false: This is the default. If you use crossref or xref fields in the four mentioned entry types, you'll get the abbreviated references in both notes and bibliography.
true: You'll get no abbreviated references in these entry types, either in notes or in the bibliography.
notes: The abbreviated references will not appear in notes, but only in the bibliography.
bib: The abbreviated references will not appear in the bibliography, but only in notes.
none: This switch is special, allowing you with one setting to provide abbreviated references not just to the four entry types mentioned but also to book, bookinbook, collection, and proceedings entries, both in notes and in the bibliography.

This option can be set either in the preamble or in the options field of individual entries. For controlling the behavior of book, bookinbook, collection, and proceedings entries, please see booklongxref, above, and also the documentation of crossref in section .

nameaddonsep =space This option sets the punctuation which appears before the nameaddon field in all entry types except customc. You can set it globally, per type or per entry, using one of the six following keys:

space = \addspace. This is the default.
none = no separator at all. It presumes that you will include one in the nameaddon field itself.
colon = \addcolon\addspace.

```
comma    = \addcomma\addspace.
period   = \addperiod\addspace.
semicolon = \addsemicolon\addspace.
```

Cf. nameaddon and nameaddonformat in section 4.4.3.

nodates=true   This option means that biblatex-chicago automatically provides \bibstring{nodate} in any "circa" or "uncertain" date specification where the user has also turned off the printing of brackets around the date using the nodatebrackets or noyearbrackets options (section 4.4.3; 14.145). If you set nodates=false either in your preamble (for global coverage or for specific entry types) or in individual entries then the package will simply print the unbracketed date specification in this context. See table 3, below. (The bibstring expands to "n.d." in English; please note that this option works quite differently in the author-date styles.)

shorthand   This option controls the punctuation that appears before the first introduction of a short-
punct   hand field, including the shorthandintro, in long notes. The default is \addspace, but if
=space   this isn't correct for your needs, especially if you change the shorthandintro or don't want the whole phrase inside parentheses, then you can change it in the preamble or in individual entries. The accepted option keys are:

none = no punctuation at all
space = \addspace
comma = \addcomma\addspace
period = \addperiod\addspace
colon = \addcolon\addspace
semicolon = \addsemicolon\addspace
emdash = \addthinspace\textemdash\addthinspace
endash = \addspace\textendash\addspace

You can, in emergencies, directly \renewcommand{\shorthandpunct} in the preamble, but it might be easier to use the none option to shorthandpunct and hand-craft solutions inside the shorthandintro fields of individual entries.

4.4.3   Style Options – Preamble

These are parts of the specification that not everyone will wish to enable. All except the sixth, seventh, and twelfth can be used even if you load the package in the old way via a call to biblatex, but most users can just place the appropriate string(s) in the options to the \usepackage {biblatex-chicago} call in your preamble.

alwaysrange   Biblatex-chicago now implements biblatex's enhanced date specifications, one part of which is the presentation of decades and centuries not as year ranges but as localized strings like "19th c." or "1970s." The alwaysrange option set to true, either in your preamble or in individual entries, simply tells the package to present the year range instead. This allows you to use the efficient enhanced notations in the date field ({18XX} or {197X}) without the localized strings appearing, should you require it. The two options centuryrange and decaderange limit the same effect to centuries and decades, respectively. Please see table 3.

annotation   At the request of Emil Salim, I included in biblatex-chicago the ability to produce annotated bibliographies. More recently, Moritz Wemheuer brought to my attention a Stack-Exchange question which suggested that the field might be useful in several other contexts as well, so I've modified the annotation option to allow the field to appear in the bibliography (=bib or =true, the default, if no string is given), in long notes (=notes), in both (=all), or in neither (=false). You can now set the option in the preamble both globally and per-type, and in the options field of individual entries. There are two new options (bibannotesep and citeannotesep) to allow you to choose the separator between the annotation and the rest of the entry, and also two new options (formatbib and entrybreak) to give you fine-grained control over the presentation of the bibliography as a whole, including an annotated one. Please have a look at the documentation for the latter two options just below, for the former two options in section 4.4.2, and for the annotation

field on page 28. Please also see biblatex.pdf § 3.13.8 for details on how to use external files to store annotations.

blogurl   As a convenience feature, this option, available only in the preamble, allows review entries to inherit a url from article entries. The main use, as the name suggests, is when you want to provide the same url for a blog comment as you have for a blog post. You'll need an extra LaTeX- Biber - LaTeX run to make sure any changes to this option take effect.

casechanger   Like mcite and natbib, this is a standard biblatex option which biblatex-chicago simply passes through to that package. In biblatex it defaults to auto, but there were, and possibly still are, cases when the old latex2e case-changing code can work around some bugs when using, for example, citation commands inside fields that have a case-changing element as an automatic part of their formatting (note, titleaddon, type). Cf. the \citeincite command in section 4.3.2, and section 3.1.1 of biblatex.pdf.

centuryrange   This option works just like alwaysrange, above, but only affects century presentation, not decade. Cf. table 3.

cmsbreakurl   The *Manual* gives fairly specific instructions about breaking URLs across lines (14.18), so I have attempted to implement them by tweaking biblatex's default settings, which are found in biblatex.def. In truth, I haven't succeeded in getting biblatex flawlessly to follow all of the *Manual's* instructions, nor do the changes I have made work well in all circumstances, being particularly unsightly if you happen to be using the ragged2e package. For these reasons, I have made my changes dependent on a package option, cmsbreakurl, which you can set in your preamble. I have placed all of this code in biblatex-chicago.sty, so if you load the package with a call to biblatex instead, then URL line breaking will revert to the biblatex defaults. See cms-notes-sample.pdf for a lot of examples of what URLs look like when the option is set, and also section 4.5.1, below.

compresspages   When set to true, any page ranges in your .bib file or in the postnote field of your citation commands will be compressed in accordance with the *Manual's* specifications (9.61). Something like 321--328 in your .bib file would become 321–28 in your document. See the pages field in section 4.2, above. Please note that the code for this is in biblatex-chicago.sty, so if you load the package with a call to biblatex instead then you'll get the default biblatex compression style.

datamodel   This is the standard biblatex option for loading the named data model file (excluding its .dbx extension). After a request by Philipp Immel, you can now set this option when you load the Chicago styles with \usepackage{biblatex-chicago}, and it will be passed through properly to biblatex itself. Cf. biblatex.pdf § 4.5.4.

decaderange   This option works just like alwaysrange, above, but only affects decade presentation, not century. Cf. table 3.

delayvolume   The presentation of volume information in the notes & bibliography style is complicated (*Manual*, 14.116–22). Depending on entry type and on the presence or absence of a booktitle or a maintitle, volume data will be presented, in the bibliography, either before a maintitle or after a booktitle or maintitle, that is, just before publication information. This, so far, is handled for you automatically by biblatex-chicago-notes. In long notes, the same options apply, but it is also sometimes better to place volume information *after* the publication information and just before any page numbers, so I have included this option, which you can set either for the whole document or on a per-entry basis, to allow you to move volume data to the end of a long note. Please note that this doesn't affect any volume data printed *before* a maintitle, but only data that would, without this option, be printed *after* a booktitle or maintitle. Cf. also \postvolpunct, below.

entrybreak   This option only makes sense when used in conjunction with the annotenp switch to the formatbib option (below). The latter allows LaTeX to break a page inside an annotation field printed without starting a new paragraph, and it does so by allowing such a break in the entry only after a set number of lines, by default set to 3. The idea is that most bibliography entries will fit within 3 lines, so the break would generally be somewhere inside the annotation. If your document needs a value different from 3, provide the integer using the entrybreak option in your preamble. Some experimentation may be needed to find the optimum number for a given document.

footmarkoff Although the *Manual* (14.24) recommends specific formatting for footnote (and endnote) marks, i.e., superscript in the text and in-line in foot- or endnotes, Charles Schaum has brought it to my attention that not all publishers follow this practice, even when requiring Chicago style. I have retained this formatting as the default setup, but if you include the footmarkoff option, biblatex-chicago-notes will not alter LaTeX's (or the endnote package's) defaults in any way, leaving you free to follow the specifications of your publisher. I have placed all of this code in biblatex-chicago.sty, so if you load the package with a call to biblatex instead, then once again footnote marks will revert to the LaTeX default, but of course you also lose a fair amount of other formatting, as well. See section 4.5.1, below.

formatbib The *Manual* in fact says very little about formatting issues in bibliographies, e.g., whether to break entries across pages and whether to allow widows and orphans (single lines at the start or end of a page). A quick and non-scientific survey of publications issued by the University of Chicago Press suggests that actual practices are extremely varied, so I've tried to provide a number of choices for users of biblatex-chicago, most of them available as keys to the new formatbib preamble option, but a few of them also involving settings of the bibannotesep and the entrybreak options. The keys of formatbib are as follows:

> **max:** This is the biblatex default, so if you don't set formatbib at all it's what you get. It provides maximal intervention, disallowing entries broken across pages, including even when an entry includes a lengthy annotation.
>
> **min:** This allows page breaks just about anywhere, including inside entries, and it also allows widows and orphans, so it will usually provide the most efficient use of available space on the pages of your bibliography.
>
> **minwo:** This is like min, but discourages widows and orphans.
>
> **annote:** This option treats annotations separately from bibliography entries, allowing them to be broken across pages while the entry itself won't be. It is intended for use with, and will only properly work with, bibannotesep set to one of the modes that start a new paragraph for the annotation, to wit, par, vpar (the default), parbreak, or vparbreak. See below for the meaning of the "break" options here.
>
> **annotenp:** This option attempts to treat annotations separately from bibliography entries in those settings of bibannotesep which don't involve starting a new paragraph. It works by setting the number of lines in an entry after which page breaking is allowed. By default entries will only break after 3 lines, the idea being that most bibliography entries fit into three lines, so at that point you're likely to be inside the annotation, but you can set the entrybreak option to any integer that works for your reference apparatus.

Please note that there is one possible break point that isn't directly addressed by these options, that is, the one between the main entry and the annotation when that annotation starts a new paragraph. If you set bibannotesep to par or vpar, then LaTeX will try very hard not to break between entry and annotation, ensuring that the annotation at least starts on the same page as its entry. If you use parbreak or vparbreak, LaTeX is positively encouraged to break a page there, as is usual between paragraphs.

You can of course ignore the formatbib option and provide your own settings. Biblatex uses the \bibsetup command which you can renew in your preamble. You can find a nice commentary on the default values set by the package in the file biblatex.def, which you'll find in the main biblatex directory of your TeX distribution.

genallnames This option affects the choice of which names to present in the genitive case when using the \gentextcites command. Please see the documentation of that command in section 4.3.2, above.

inheritshorthand Setting this option to true allows child entries to inherit the shorthand and shorthandintro fields from cross-referenced parent entries. This in turn allows abbreviated references to the parent entry to use the shorthand instead of the usual and merely short citation, thus allowing for extra space savings. There are several other steps required to make this all function smoothly, so please see the documentation of the **shorthand** field in section 4.2, above.

journalabbrev This option controls the printing of the shortjournal field in place of the journaltitle field in notes and bibliography. It is false by default, so as shipped biblatex-chicago-notes will

silently ignore such fields, but you can set it, either in the preamble or in individual entries, to one of three other values: true prints the abbreviated form in notes and bibliography, notes in notes only, and bib in the bibliography only. Please note that in periodical entries the title and shorttitle fields behave in exactly the same manner. For more details, see the documentation of shortjournal in section 4.2, above.

juniorcomma   The *Manual* (6.43) states that "commas are not required with *Jr.* and *Sr.*," so by default biblatex-chicago has followed standard biblatex in using a simple space in names like "John Doe Jr." Charles Schaum has pointed out that traditional BIBTEX practice was to include the comma, and since the *Manual* has no objections to this, I have provided an option which allows you to turn this behavior back on, either for the whole document or on a per-entry basis. Please note, first, that numerical suffixes (John Doe III) never take the comma. The code tests for this situation, and detects cardinal numbers well, but if you are using ordinals you may need to set this to false in the options field of some entries. Second, I have fixed a bug in older releases which always printed the "Jr." part of the name immediately after the surname, even when the surname came before the given names (as in a bibliography). The package now correctly puts the "Jr." part at the end, after the given names, and in this position it always takes a comma, the presence of which is unaffected by this option.

nameaddon   This option determines where and when the nameaddon field will be printed. There are seven possible values, the first four of which are available globally, per type, and per entry, with the last three only applicable globally:

   all: This is the default; if an entry has a nameaddon, it will appear in both long notes and in the bibliography.
   none: The field will appear neither in the bibliography nor in long notes.
   bib: The field will appear only in the bibliography.
   cite: The field will appear only in long notes.
   first: Philipp Immel requested this as a way to provide an author's dates in the nameaddon field and only have them printed the first time that author appears in the bibliography. A sequence of consecutive long notes citing works by the same author will be treated the same way. The code tests for identical nameaddon fields in works by identical authors, so other sorts of nameaddon will be printed as usual.
   bibfirst: Like first, but will not print the nameaddon field in long notes.
   citefirst: Like first, but will not print the nameaddon field in the bibliography.

   Cf. nameaddonformat just below, and nameaddonsep in section 4.4.2.

nameaddon-format   This option, available globally, per type, and per entry, allows you to change the format of the nameaddon field on the fly, so its value should be a field format that biblatex understands. This includes standard formats like parens, brackets or emph, and also custom formats that you provide in your preamble using \DeclareFieldFormat, in case the standard ones aren't adequate. If you don't define this option, then the usual defaults apply, that is, no formatting in online, review, and suppperiodical entries, as well as in misc entries with an entrysubtype, while square brackets surround the field in all other entry types with the exception of customc, which has its own rules and ignores this option. Cf. nameaddonsep in section 4.4.2.

natbib
mcite   These are the standard biblatex options, which formerly required slightly special handling when you loaded the Chicago style with \usepackage{biblatex-chicago}. Both the forms natbib and natbib=true (mcite & mcite=true) should now work.

nodatebrackets
noyearbrackets   When you use biblatex's enhanced date specifications to present a "circa" date ({1956~}), an uncertain date ({1956?}), or one that is both at the same time ({1956%}), the date that by default will appear in your documents will have square brackets around it. This accords with the *Manual's* instructions concerning such dates (14.145), but that section also includes an alternative form, where the guessed at date appears, without brackets, after the \bibstring{nodate}, e.g., "n.d., ca. 1750." These two package options, which may appear in the preamble either for the whole document or for specific entry types, or in individual entries, allow you to control when these brackets will appear, while the nodates option, set to true by default, decides whether to print \bibstring{nodate} before the date. In truth, users of the notes & bibliography style will probably only ever

need `nodatebrackets`, which controls most of the dates that will appear in your documents, with the exception of dates in some article, review, and periodical entries without an `entrysubtype`, which are governed by `noyearbrackets`. (The distribution is different in the author-date styles, so it's impossible to do without both options.) Cf. table 3.

noibid   At the request of an early tester, I have included this option to allow you to turn off the *ibidem* mechanism that `biblatex-chicago-notes` uses by default. Some publishers, it would appear, require this. Setting this option will mean that instead of the *ibidem* mechanism you'll get the short note form. Please note that the 17th edition no longer recommends the use of "*ibid.*" at all (14.34), so depending on the state of the `useibid` option, below, what you'll be turning off may well no longer be the appearance of *ibid.* itself. The option is settable globally, per type, or per entry, so that fine-grained control of individual citations is now possible without the use of specialized citation commands, though these commands are still available in a pinch. See section 4.3.2.

notitle   When citing sources from antiquity (using the `classical` entrysubtype), the *Manual* (14.244–5) recommends using just the author in short citations if only one title by that author has come down to us, making the identification of the work unambiguous. I previously suggested using a command like `\citeauthor` to achieve this, but Tobias Becht suggested that a less clumsy method would be better, so I've provided the notitle option, settable by entry type and also in the `options` field of specific entries. The option has no effect whatever in long notes. Cf. herodotus:wilson.

omitxrefdate   As part of the abbreviated cross-referencing functionality for book, bookinbook, collection, and proceedings entries, I have thought it helpful to include, in the abbreviated references only, a date for any title that's part of a `maintitle`, though not for those that are only part of `booktitle`. If these dates annoy you, you can use this option to turn them off, either in the preamble for the document as a whole or in the `options` field of individual entries. Cf. harley:ancient:cart, harley:cartography, and harley:hoc; and crossref in section 4.2, above.

ordinalgb   This option, which only affects users of the british language, restores the previous package defaults, printing the day part of a date specification as an ordinal number: 26th March 2017. The new package default prints 26 March 2017, which is more in keeping both with standard British usage and with the recommendations of the *Manual* (6.38). The option is available only in the preamble.

postnotepunct   Several users, most recently David Gohlke, have requested a way to alter the punctuation
(experimental)   that appears just before the postnote argument of citation commands, usually, but perhaps not always, to allow citations to fit better into the flow of text. This punctuation is a complex issue in the *Manual*, and I've attempted to make `biblatex-chicago` follow the specifications closely. Still, as a first stab at enabling the greater flexibility in punctuation that some have requested, I have introduced the `postnotepunct` package option. Set to `true`, it allows you to start the postnote field with a punctuation mark (. , ; :) and have it appear as the `\postnotedelim` in place of whatever the package might otherwise automatically have chosen. Please note that this functionality relies on a very nifty macro by Philipp Lehman which I haven't extensively tested, so I'm labeling this option experimental. Note also that the option only affects the postnote field of citation commands, not the pages field in your .bib file.

seriesabbrev   This option controls the printing of the shortseries field in place of the series field in book-like entries in notes and bibliography. It is `false` by default, so as shipped `biblatex-chicago-notes` will silently ignore such fields, but you can set it, either in the preamble for the whole document or for specific entry types, or in individual entries, to one of three other values: `true` prints the abbreviated form in notes and bibliography, `notes` in notes only, and `bib` in the bibliography only. For more details, see the documentation of shortseries in section 4.2, above.

short   This option means that your text will only use the short note form, even in the first citation of a particular work. The *Manual* (14.19) recommends this space-saving format only when you provide a *full* bibliography, though even with such a bibliography you may feel it easier for your readers to present long first citations. Tobias Becht pointed out that

the *Manual* (14.242–54) envisages using short citations for ancient, medieval, and Renaissance works even in the first reference, and this without regard to whether you do so with other works, so now you can set the short option for a whole document, for specific entry types, and for specific entries. If you do use the short option, remember that there are several citation commands which allow you to present the full reference in specific cases (see section 4.3.2). If your bibliography is not complete, then you should probably not set this option globally.

shortextra-
field
shortextra-
format
shortextra-
punct

N. Andrew Walsh has remarked that it is quite possible for documents to cite works that, though perfectly distinguishable in their long form, end up looking identical in short notes; multiple performances of the same work by the same artist, for example, might end up producing such a situation. (In online and review entries using commenton relatedtype this situation is so endemic that I've set a default method of disambiguating short notes there, though you can still override it with the following options. See section 4.2.1, above.) While the use of a shorthand field could provide some sort of remedy, he requested a way to disambiguate short notes by adding a user-configurable field to the note, thereby keeping it both short and unique without the need to consult a list of shorthands. The *Manual* (15.29) itself provides just such a mechanism in the author-date specification, so I've added one to the notes & bibliography style, as well. It consists of the standard biblatex option labeltitle, now set to true by default, along with three package options for the user to configure. All three of these options are settable for the whole document, for individual entries, or for individual entry types.

The first is shortextrafield, which *must* be set in order for the mechanism to print anything at all. You should set this option to the name of the field you wish to be printed in addition to the author and labeltitle. (Possibilities include, but are not limited to, the 4 *date fields and the 4 *time fields, the latter of which will print the appropriate *date and the *time.) By default, it will be printed after the latter, separated from it by a comma. You can manually define this punctuation by setting the shortextrapunct option to one of none, space, comma, period, colon, or semicolon. You can also enclose the extra field in parentheses or square brackets by setting the shortextraformat option to parens or brackets.

shorthand-
first

User laudecir requested a simpler way to print the shorthand even in the first citation of a source, simpler, that is, than remembering to use the \shorthandcite command. You can set this option to true either in the preamble or in individual entries.

shorthandfull

Kenneth Pearce has suggested that, in some fields of study, a list of shorthands providing full bibliographical information may replace the bibliography itself. This option prints this full information in the list of shorthands, though of course you should remember that any .bib entry not containing a shorthand field won't appear in such a list. Please see the documentation of the **shorthand** field in section 4.2 above for information on further options available to you for presenting and formatting the list of shorthands.

shorthandibid

Chris Sparks pointed out that biblatex-chicago-notes would never use *ibid.* in the case of entries containing a shorthand field, but rather that consecutive references to such an entry continued to provide the shorthand, instead. The *Manual* isn't, as far as I can tell, completely clear on this question. In 14.244, discussing references to works from classical antiquity, it states that "when abbreviations are used, these rather than *ibid.* should be used in succeeding references to the same work," but I can't make out whether this rule is specific to classical references or has more general scope. Given this ambiguity, I don't think it unreasonable to provide an option to allow printing of *ibid.* instead of the shorthand in such circumstances, though the default behavior remains the same as it always has.

shorthand-
intro

Fr. Norbert Keliher requested a way to turn off the printing, in the first citation of a work, of the introduction of a shorthand that will appear in subsequent citations. (A possible use case is when that shorthand is so well known that it needs no introduction.) By default, biblatex-chicago prints a notice — (hereafter cited as) — when no shorthandintro field is present, the latter field allowing you to modify the notice but not eliminate it. The new shorthandintro option is available globally, per-type, and per-entry, and has three possible values:

none: This setting turns off the notice entirely. The shorthand field will simply appear in subsequent citations with no introduction.

short: This setting provides the shorthand alone in parentheses at the end of the first citation. You can get a similar effect by providing a shorthandintro field which only contains the shorthand wrapped in parentheses.

full: This is in effect the default setting, but you would only need to use it in a document where one of the other two settings was in force either globally or per-type. In chicago-notes.cbx I have set this value for the jurisdiction, legal, and legislation entry types to fulfil the *Bluebook's* specification even when a user changes it for other sorts of entry. This requires you to alter it specifically for these types should you wish to depart from the *Bluebook* guidelines. Cf. section 6, below.

N.B. The list of shorthands will always still be available to you as an explanatory tool in the absence of any introductions to individual shorthands. Please see the documentation of the shorthand and shorthandintro fields in section 4.2.

strict   This still-experimental option attempts to follow the *Manual's* recommendations (14.41) for formatting footnotes on the page, using no rule between them and the main text unless there is a run-on note, in which case a short rule intervenes to emphasize this continuation. I haven't tested this code very thoroughly, and it's possible that frequent use of floats might interfere with it. Let me know if it causes problems.

urlnotes   Stefan Björk, for specialized reasons, requested a way to turn off the printing of url, doi, and eprint information in notes but not in the bibliography. As it's possible this might be of more general usefulness, I've provided a new option. You can set it to false either in the preamble or in individual entries, but please note that it does not apply to online entries.

useibid   In a change to previous recommendations, the 17th edition of the *Manual* "discourages the use of *ibid.*," preferring instead a shortened reference with only the author's name (14.34). Biblatex-chicago now implements these recommendations by default, including the repetition of page references even when they're exactly the same as the previous note. If you prefer to continue using *ibid.*, then set useibid=true in your document preamble either globally or per type, or else in the options field of individual entries, and you'll get the traditional behavior (To be perfectly accurate, in entries with classical entrysubtype and which have a shortauthor, the *ibid.* string is suppressed unless you set useibid=true in the options field of the entry itself. The global setting will in this case be ignored.)

xrefurl   Stefan Björk pointed out that when, using the longcrossref or booklongxref options, you turn on the automatic abbreviation of multiple entries in the same (e.g.) collection or mvcollection, you could entirely lose a url that might be helpful for locating a source, as the abbreviated forms in notes and in the bibliography wouldn't include this information. Setting this option to true either in the preamble or in individual entries will allow the url, doi, or eprint field to appear even in these abbreviated references.

4.4.4 Back References: The `noteref` Option

Biblatex has always provided the backref option, which prints, in the bibliography, those pages on which individual works have been cited. The *Manual* (14.31) recommends another, related system, which involves, at the end of short notes, cross-references to the note where the reader can find the full, long citation of the same source, "especially in the absence of a full bibliography." The general idea is that, where a short note is "far" from the long citation, a back reference to that long note may prove "helpful." The recommended format is something like this: (see chap. 1, n. 4). The previous release of biblatex-chicago provided something similar only for certain subsets of material from the *Bluebook* guidelines for legal citations. That provision is very basic and has a different rationale, so it remains both unaltered and entirely separate. With this release, in the notes & bibliography style only, I have provided Chicago-style back references for all other entry types, enabled through the noteref preamble option. (Cf. cms-noteref-demo.pdf for a brief introduction.)

Before embarking on a description of this new option, and its many sub-options, I would like to point out that biblatex provides a number of mechanisms designed to help readers

navigate long documents. I have made many improvements to the `biblatex-chicago` hyperref interface, so electronic documents can, at your discretion, contain clickable links from short notes to long notes or, in a document with all short notes, from such notes to bibliography entries. (The `noteref` mechanism cooperates well with `hyperref`, and therefore can add another layer of links to those already available.) `Biblatex` also offers the `refsection` and `citereset` preamble options, which allow you to choose how its citation trackers behave. Using these you can, for example, always have a long note appear for a given source at its first appearance in a chapter or a section, something which the *Manual* recommends in any case, and which may mean that your short notes are never too "far" from a longer citation. (See biblatex.pdf, § 3.1.2.1.)

noteref All the same, sometimes chapters or sections can get rather long, or a too-frequent reappearance of the long form may not be optimal for your work, so in such situations the `noteref` option may well prove useful. Its general principle is this: if a short note appears on the same page as its corresponding long note, or on the same double-page spread for `twoside` mode, then nothing will appear. Similarly, if a short note appears on the same page or double-page spread as a previous short citation of the same source which *does* have a `noteref`, then this subsequent short citation will once again *not* present any `noteref`. (This behavior is configurable — please see below.) If a back reference is to be printed, then the value of the `noteref` option determines what it will look like. Its six possible values are:

> none: This is the default, and will always produce a back reference like this: (see n. 1). It may well be useful when you are using the `citereset` or `refsection` options and know that any short note will always be in the same chapter or section of the text as the full reference to which it points.

> page: This will always produce a back reference like this: (see p. 1, n. 1). It can be a tidy way of directing the reader, as page numbers are usually simpler to track than sections or chapters. It's also a good setting if you've set the LaTeX `secnumdepth` counter so that sections aren't numbered.

> chapter: This is the example provided by the *Manual*, and as implemented here it produces two different sorts of back reference. If the short note is in the *same* chapter as the long note to which it points, then by default it will only mention the note number, as with the `none` switch, above. If the long reference is in a different chapter, then it prints like so: (see chap. 1, n. 4). All of the options that name an organizing division of the text work the same way in footnotes, i.e., only when the short note and the long note to which it refers are in different `parts`, `chapters`, `sections`, or `subsections` will the actual division type appear in the `noteref`. If fullnoterefs you want the longer form in *all* of your `noteref`s, you can set the biblatex-chicago option `fullnoterefs` to `true` when loading the package. In endnotes, depending on which options you've chosen for presenting them, you may well never get the short version of the back reference. Please see the details starting on page 85, below.

> section: This key is particularly intended for documents, like the standard LaTeX article class, which don't offer chapters, but rather start their divisions at the section level, but it's perfectly usable even in a document that also uses chapters. Assuming the short note and its long antecedent are in different sections, the `noteref` will look like so: w/o chapter (see § 2, n. 6), w/ chapter (see § 1.2, n. 6).

> subsection: I'm not sure there's any need for this key, but I include it for the sake of completeness. It's usable in documents both with and without chapter divisions, and assuming the short note and its long antecedent are in different subsections the `noteref` will look like so: w/o chapter (see § 3.2, n. 5), w/ chapter (see § 2.3.2, n. 5).

> part: This is, I suspect, even less likely to be useful than `subsection`, but assuming the short note and its long antecedent are in different parts the back reference will look like so: (see pt. I, n. 4). You'll need to be careful that note numbering is continuous across chapters for this to work correctly, otherwise the plain note

number might well be ambiguous. Also, if you'd like the part number not to be roman, you can try putting this in your preamble: `\let\cmsnrpart\relax`.

Several comments are in order, before moving on to the onerous details. In case it's not already clear, the `noteref` option is only relevant if your document includes full notes, that is, if you aren't using the `short` option. Even in documents that use long notes, it can occasionally happen that the `noteref` code won't be able to find a full citation of a particular source. In this case, no back reference will appear, and you will find a warning in your .log file informing you about it. (If you combine `short` and `noteref`, you'll see a lot of such warnings.) It can happen that even the first citation of a particular source appears in a somewhat abbreviated form, as when multiple contributions to the same `collection` are present in your reference apparatus. The `noteref` will point to this abbreviated first citation all the same, given that it is at least somewhat more informative than a short note. Similarly, a `noteref` from a `collection` may well point to the long citation of an essay from that collection, as that long citation will contain all the details of the collection, too. I hope this doesn't prove too surprising. I should also clarify that all of the strings in the `noterefs` as printed above are localized, so should adapt to your document language reasonably smoothly, if not entirely idiomatically. Finally, the code assumes that the standard LaTeX counters for parts, chapters, and sections are available, which I believe is almost universally the case even for classes and styles that redefine a lot of the relevant functionality, but I confess I haven't tested `noteref` at all extensively against the possibilities offered by CTAN, so please do let me know if something breaks for you.

noterefinterval   I mentioned above that the gap between appearances of a `noteref` for a given source was configurable. What I had in mind was this option, along with four new citation commands which I discuss below. The `noterefinterval` preamble option allows you to define the number of references that must have intervened since the last `noteref` before another to the same source will appear. If you judge that your readers don't need a pointer on every new page but only after a certain number of other citations have passed, you can set this to a number higher than zero (the default). You can't, currently, use this mechanism to make new pointers appear on the same page as previous ones, but you can spread them out if they appear too frequently for your tastes. Also, the counter that this option uses is `instcount`, which will be incremented not only by new references but also, e.g., by uses of the `related` functionality to extract data from other entries. A value of 15 may not delay a reappearance for exactly that many notes, so you will need to experiment a little to find a value that suits your document.

\shortrefcite
\shorthand-
refcite
\shortcite*
\shorthandcite*   If you require more fine-grained control over the spacing between `noterefs`, or indeed if you want them to appear more frequently than the previous mechanisms allow, then these new citation commands will allow you to do so, though perhaps without maximum convenience. (I suppose that it would be safest to introduce these commands into your documents at quite a late stage in their preparation.) The first two commands present, respectively, short notes and shorthand notes where the `noteref` will absolutely be printed (unless, of course, no full citation can be found). The second two commands prevent the printing of the `noteref`, no matter where the resulting note appears. All of them will need enclosing in a `\footnote` command if you want them to appear in one, as I've provided suppressnoterefs   only the most general form of each. I've also, after a user request, provided the suppress-noterefs option, settable per entry type and per entry, which prevents any `noteref` being printed for the entries concerned. This may be more convenient than the specialized citation commands.

Zero Sections

The LaTeX sectioning counters all start from zero, so if you put a note into material occurring *before* the first `\part`, `\chapter`, `\section`, or `\subsection` command then any back reference to this citation will, by default, present that zero (or zeros). This may, in fact, be exactly what you want, in which case you can ignore the following options. If you don't want a zero to appear in your `noterefs`, you can either make sure no citations occur in contexts that will produce them, or you can use a combination of the next three options to hide them.

noterefintro   This option is special in that it handles only zeros that occur in the *first* position in a sectioning identifier, e.g., § 0.x.x or chap. 0. It's possible that this zero represents some sort

of introductory material before, e.g., the numbered chapters appear. If you say `noteref-intro=introduction`, then instead of (see chap. 0, n. 1), you'll have (see intro., n. 1). If the value of the option is a `\bibstring` known to biblatex, then it will appear in localized, and possibly abbreviated, form, as above. If the section title you want isn't a known `\bibstring`, you can either define a new one for your language in your preamble, or you can just set the option to whatever it is you want to appear in such `noterefs`. Both will work, particularly because you don't need to worry too much about capitalization because the word always appears after the `\bibstring{see}`.

Now, it's perfectly possible for an introduction to have numbered sections of its own, so a citation there may produce a back reference like § 0.1 or § 0.1.0. The rules are: 1. any back reference which is all zeros will just print the `noterefintro` string alone, assuming you've provided one; 2. any back reference that has the zero *only* in the first place will print something like (see intro., § 1, n. 1); 3. a back reference of the form 0.x.0 or 0.0.x — this can only occur if `noteref=subsection` and the document class provides a `\chapter` command — such a reference will either print the zero after the `noterefintro` string, or you can use the `pagezeros` option, to which I turn.

pagezeros   This boolean option deals with the problem of zero sections by changing such back references, and only such back references, to behave as though `noteref=page`. In the absence of a noterefintro *all* such zero citations will be so treated, but if both options are set then zero sections with the zero in the first position of the identifier will fall first under the jurisdiction of the `noterefintro` option, only then turning to `pagezeros` if there's a zero remaining that hasn't yet been eliminated by the first option's rules. Some examples:

> **Without noterefintro:**
> Any zero —> (see p. 3, n. 1)
> No zero —> (see § 1.2.3, n. 1)
>
> **With noterefintro:**
> 1.2.3 —> (see § 1.2.3, n. 1)
> 0.0.0 —> (see intro., n. 1)
> 1.1.0 —> (see p. 24, n.1)
> 0.1.0 —> (see intro., p. 2, n. 1)

As you see, this produces a mixed system of back references, so you'll need to decide whether you and your readers might still find it acceptable.

hidezeros   This boolean option attempts, with varying degrees of success, to disguise the zeros in section identifiers without mixing different sorts of back reference in one document. Unlike `pagezeros`, it will never modify identifiers where the first number is zero. It leaves all such identifiers to the `noterefintro` option, so if your identifier looks like 0.1.0, the second zero will still appear. The only thing you can do about it is to move the citation so that it isn't in a zero section. The rationale is that `hidezeros` places a string — by default "pref." — *inside* the section identifier so that a reader knows that the citation occurred in the prefatory material to a particular section. Combining this with another string for the prefatory material to a whole work is unattractive, and I've avoided it. Here are some examples of how it looks:

> 1.1.1 —> (see § 1.1.1, n. 1)
> 1.0.1 —> (see § 1.pref.1, n. 1); should this even be possible?
> 1.1.0 —> (see § 1.1.1 [pref.], n. 1)
> 1.0.0 —> (see § 1.1 [pref.], n. 1)
> 1.0 —> (see § 1.1 [pref.], n. 1)

\cmspref   The brackets and placement of the identifying string are hard coded, but you can change the string itself with a line something like this in your preamble:
`\renewrobustcmd{\cmspref}{\emph{pref.\@}}` <— **NB** the `\@` after the dot
The `hidezeros` method at least produces more uniform back references, though it perhaps sacrifices something in immediate readability in order to do so. I would be glad to entertain suggestions for other solutions.

If your documentation uses footnotes, then the guide to the `noteref` option(s) concludes here. I have, however, been determined, at least in this context, to provide for endnote

users the same features as for footnote users, mainly because the *Manual* caters equally to both. Unfortunately, because footnotes are a core part of LaTeX formats and endnotes are provided by means of extra packages, users of the latter will face some additional complications if they wish to provide Chicago-style back references. I document these forthwith.

Endnotes and `noteref`

The traditional way to provide endnotes instead of footnotes in a document is to use the endnotes package by John Lavagnino, and biblatex provides a reliable interface to that package, making it relatively simple to use and control (cf. esp. the `notetype` option in biblatex.pdf, § 3.1.2.1). The package does have its limitations — on which see more below — so I did have a look at its next-generation replacement, Clemens Niederberger's enotez. This adds all the needed functionality, I think, and is also more future proof, relying as it does on the work of the LaTeX3 project. The downside is that my TeXnical abilities fell short of being able to make it work with the `noteref` option, so I have instead created a new

`cmsendnotes.sty`   package which combines functionality from endnotes, from Ulrich Dirr's hyperendnotes, and from biblatex-chicago. If you need some functionality that endnotes doesn't provide, then you can load cmsendnotes instead. (I haven't tested any of this with the memoir class, which has its own endnote mechanism, so please let me know if it doesn't work and I'll try to have a look.) The documentation that follows should clarify when you might want to load the new package, and also the options available to get cmsendnotes to do what you want.

Before we begin, I should just point out that, as usual with biblatex, you can mix foot- and endnotes in the same document, but if `noteref`s are going to appear in both sorts of note — surely this situation is highly unlikely — then you need to be careful that they refer back *only* to long references in the *same* sort of note. A `noteref` from an endnote to a long citation in a footnote will be inaccurate, and vice versa, so careful use of the `\citereset` command (as in cms-noteref-demo.pdf) or perhaps of the biblatex `citereset` option should allow you to keep the two sorts of note distinct.

The principle to keep in mind is that back references to full endnotes point not to the place in the main text where you've cited a source, but rather to the place where that citation is actually printed, which may well be in another division of your document altogether. If you are providing endnotes at the end of each chapter, or (less likely) at the end of each section of a long article, then this means that an endnote to a later chapter or section will point to the earlier chapter or section after which the full citation was printed. This interacts very well with the endnotes package's `\theendnotes` command, which prints, and then clears, all the endnotes created up to the point at which you call it. Subsequent calls to `\theendnotes` do the same, and short notes will always have a reasonably accurate sense of where their antecedent long note has appeared, i.e., in the endnotes to a particular chapter or section. (This even has the side effect of making the zero section problem somewhat more tractable, as the back reference doesn't mind that the `\endnote` command occurs in section 1.0, but rather that the citation appears in the notes to section 1.1.)

Similarly, if your endnotes appear all together at the end of an article, then you can just use the `page` option to `noteref`, or no option at all, and the back references will be both accurate and usable (assuming the notes are all numbered consecutively, I suppose, which seems a safe assumption). The upshot is that, if you are providing endnotes in either of these scenarios, both of them envisaged by the *Manual* (14.46), and *either* if you don't need the hyperref functionality, *or* if the somewhat restricted functionality available through the endnotes package is good enough, then you can happily ignore the new cmsendnotes package entirely.

If back references are, in the scenarios discussed above, basically working for you, but you want more elaborate hyperref functionality, then you can load the cmsendnotes package without any options instead of the endnotes package. Please be aware, however, that

NB   you must load cmsendnotes *after* biblatex-chicago for it to work properly. What you'll then get by default, assuming you've loaded hyperref, are links from endnote numbers in the main text to the corresponding numbers in the endnotes section itself, and vice

versa, along with more accurate links from the back references to particular endnotes in preceding sections or chapters. There are several options available for changing the default appearance of your endnotes, four of which are package options to cmsendnotes and two of which are commands that you can redefine to your liking.

hyper

enotelinks

Two package options control the hyperref behavior of endnote numbers. They are both set to true by default, if the hyperref package is loaded. If you set the first, hyper, to false when loading cmsendnotes, there will be no hyperlinking of endnote numbers at all. If you set the second, enotelinks, which I've borrowed from hyperendnotes.sty, to false, then only endnote numbers in the main text will function as links, the numbers in the endnote sections themselves ceasing to act as such.

noheader

This package option, if set to true, stops the printing of the usual section header before the endnotes themselves, in case this might help to solve some formatting problems in your documents.

\enoteheading

This is the standard endnotes package command for defining exactly what is printed in the heading. I have slightly redefined it (for reasons I shall explain later), but you can redefine it in your preamble if you wish, and that will be respected by cmsendnotes.

blocknotes

This package option, if set to true, presents the text of each endnote as a flush-left block, i.e., without the first line being indented.

\enoteskip

This command, which was inspired by a similar provision in hyperendnotes.sty, sets the vertical space between individual endnotes. By default it doesn't change basic interline space, but you can define it in your preamble to something like \smallskip or \medskip in case you want a bit more light inside your endnote sections.

So far, then, I have discussed contexts where using cmsendnotes.sty only brings cosmetic changes to functionality which basically already works using endnotes.sty. Indeed, if you are using either of these packages in the ways already outlined, then the setting of the main noteref option defines how your back references will look, and the fullnoterefs option will still govern the chapter, section, subsection, and part values of the noteref option, just as in the general discussion above. In the methods discussed below, additional steps are required for defining how your back references will look, and the fullnoterefs option is irrelevant, as the back references will always appear in their fullest form.

fullnoterefs

The methods of endnote presentation to which I now turn involve, in the *Manual's* words, when notes to "each chapter of a book are … grouped in the end matter" (14.46). Here, you would want not only a main heading for the endnotes section but also "a subhead bearing the chapter number or title or both." It is perfectly possible to achieve the subdivision and subheading of a long endnotes section by using endnote's \addtoendnotes command in each chapter of your document, putting a sectioning command of some sort in its argument, for example. Using noteref back references in this context can be a little complicated, however, mainly because of the principle I explained above, i.e., that back references point to the place where the long note was printed, not to the section of the main document where the source was actually cited. For our purposes, this means that, unless you have set noteref to page or none, biblatex-chicago actually has to extract data from the sectioning command you've included in \addtoendnotes and, most frequently, modify that data to make it work inside a noteref back reference in way that is both consistent and useful. The cmsendnotes package tries to automate this process as much as possible so that with, in the best-case scenario, only one option given to the package the whole system can be made to work without further user intervention. Further package options can help with slightly more complicated scenarios, but if your requirements are more complex than the automatic system can provide, then there are two ways to handcraft a divided endnote section: one uses traditional \addtoendnotes functionality from the endnotes package, and the other uses new commands available from cmsendnotes. I believe the second handcrafted option to be slightly more convenient than the first, but in any case I'll start by explaining the automatic provisions, then move on to the two handcrafted options, leaving you to judge which seems best suited to your needs.

split

For the automatic subdivision of an endnotes section I have borrowed a concept, if not its implementation, from enotez, and provided the cmsendnotes option split, which has

<table>
<tr><td>

\theendnotes-
bypart

</td><td>

4 possible values: part, chapter, section and subsection. (If you don't provide a key, it defaults to chapter.) With this option set, you need to use a new command for printing the endnotes, the ungainly but I hope memorable \theendnotesbypart. When you do this, cmsendnotes does something in the background that's worth understanding. Ordinarily, in the standard endnotes package, any call to \theendnotes produces an .ent file containing all of the endnote data in the document up to that point, and proceeds to print it. Another call to \theendnotes gathers the endnote data occurring between it and the first call, overwriting the .ent file, and again printing it, and so on. Whenever you use any version of \theendnotesbypart, cmsendnotes will write one .ent file per section named by the split option, assuming that said section actually contains any endnotes. The plain \theendnotesbypart command, with no further options, proceeds to print, in sequence, *all* the .ent files in numerical order. In the first instance, then, the value of the split option decides how your endnotes are distributed to different .ent files. These files are named after the main document suffixed with the number of the section, e.g., jobname1.ent for chapters or jobname1.3.ent for sections.

</td></tr>
</table>

In the second instance, the split option governs what the subheaders of your endnotes section will contain. The main header is produced by a \section* command, and by default it will read **Notes**. The subheaders are produced by \subsection* commands, and will take their name from the split option and their number from the number of the .ent file currently being processed: **Chapter 1** … **Chapter 2**. The headers are localized, assuming you're using a language supported by biblatex-chicago. Even if you're not using noteref back references in your document, this mechanism can still provide a convenient means of subdividing an endnotes section.

If you are using noteref, then the value of that option leads to two possible outcomes. If set to page or none, any back references will point to full notes by page plus note number or just by note number, as usual, bearing in mind that the page involved is where the full note was printed, not where it was cited in the main body of the text. If set to any of the other possible sections of your document, then the split option has a third function, which is to provide the (localized) string for the back reference itself — (see chap. 1, n. 2) — which will refer to a subsection of the endnotes section named **Chapter 1** rather than to the actual first chapter of the main document. This setup will usually involve setting split to the same value as the noteref option itself, but if they differ, and noteref isn't page or none, then split takes precedence and governs the appearance of the back reference.

Let's say, then, that your document is in English and you've set the biblatex-chicago option noteref=chapter, and the cmsendnotes option split, then what you can expect to see, when you use \theendnotesbypart, is something like this, subject to the usual rules for the appearance or non-appearance of noterefs, and remembering that in this context, as noted above, all noterefs will appear in their long form:

Notes
Chapter 1
1. Book.
2. Article.
3. InCollection.
Chapter 2
1. InCollection (see chap. 1, n. 3).
2. BookInBook.
3. Article (see chap. 1, n. 2).
Chapter 3
1. BookInBook (see chap. 2, n. 2).
2. Book (see chap. 1, n. 1).

<table>
<tr><td>

Zero Sections

</td><td>

It won't, unfortunately, always be this simple, but it may be a comfort to know that some of the complications are the same as those faced by users of noteref with footnotes, in particular the zero section problem. To deal with this issue you use the same options to biblatex-chicago as you would for footnotes, with one difference. In the case of split endnotes, the code has to handle the zeros both in the \subsection* names *and* in the back references, which means that the pagezeros option is no longer relevant, as it can't do the right thing in section names. This leaves the noterefintro option for hiding zeros

</td></tr>
</table>

that start a section number, and the `hidezeros` option for zeros anywhere else. These `biblatex-chicago` options work here just as described in the footnote section above, but with one or two additional caveats.

\cmsintrosection

First, I probably shouldn't have bothered trying to implement the combination of `noterefintro` and `hidezeros`, as any sections of a zero chapter in your document will appear by default with zeros intact in the text itself, unless measures are taken. I did do this thing, however, and part of the implementation is a command `\cmsintrosection`, which provides the identifying string for subsections of the introduction. It is set by default to §, and though you can redefine it in your preamble, please remember that it will appear as such in both `\subsection*` names and `noterefs`.

\introduction-
name
\forewordname
\sectionname
\subsectionname

Indeed, it is the need to cater for two quite distinct contexts that makes the automatic provision of `noteref` back references in a divided endnotes section surprisingly tricky. You need one mechanism to take `chapter` and turn it into Chapter, and quite another to turn it into chap., bearing in mind that `\bibstrings` don't work outside of the reference apparatus, and therefore not in `\subsection*` names, though obviously they're perfect for the back references themselves. My solution is to borrow a principle from babel, which provides for its languages commands like `\prefacename` and `\chapter-name`, which print the localized version of the term, usually capitalized. In the .lbx files that come with biblatex-chicago I have added `\introductionname` and `\forewordname`, along with `\sectionname` and `\subsectionname`, so at least the most common types of prefatory material, when provided to the `noterefintro` option, should work properly both in headings and in `noterefs`, and across languages.

So, another caveat. Should you wish to provide a `noterefintro` value that isn't a standard `\bibstring` or doesn't have a corresponding `\[value]name` command, then it may not work well for you in one or other of the two contexts in which it can appear. The code does test whether the bibstring and command exist, and it will capitalize anything in section headers, but otherwise you can just cunningly craft a string that's good in both contexts or you can define a `\[value]name` command and a new `\bibstring` for the value in your language, both in your preamble.

subheadername

The same caveat applies if you want to supply your own name for the `\subsection*` commands that divide up the general endnotes `\section*`. Let's say for some reason you want subsections called **Further Remarks** instead of **Chapter**. Strings of more than one word are difficult for the code to manage correctly, so instead you could include in your preamble lines looking approximately like this:

```
\NewBibliographyString{furthrem}
\DefineBibliographyStrings{american}%
    {furthrem = {furth\adddotspace rem\adddot},}
\def\furthremname{Further Remarks}
```

Then you could set subheadername=furthrem in the options to cmsendnotes and you'll get what you want. It's not wildly convenient, but it's slightly less typing than the hand-crafted options I discuss below, though for anything more complicated you'll probably need those options.

headername

Before I move on to the handcrafted methods, I should point out two more cmsendnotes options. The first, `headername`, sets the name of the main `\section*` command at the start of the endnotes section. It defaults to the usual endnotes package command `\notes-name`, which gives **Notes** in English. I have kept this separate from the standard `\enote-heading` because it needs slightly different treatment in a divided endnotes section. If the definitions I have provided of `\notesname` in the .lbx files that come with biblatex-chicago aren't to your liking, you can provide a string here instead, which is simpler to do because it shouldn't be turning up in any `noterefs`. If you'd like to redefine any of the `\*name` commands, the best place to do so is very near to where you actually print the endnotes, where it can override the definitions in the .lbx files (or in babel's files). Remember, too, that you can use the `noheader` option to turn off the printing of this header if you just want to provide your own sectioning command instead.

runningname

This `cmsendnotes` option controls the text that appears in running headers in the endnotes section of your document, should you be providing them. I have followed the style

of the endnotes package, so that the default reads something like: *NOTES TO CHAPTER 1.* The section name and number are controlled by the other options already discussed, but the "Notes to" part is controlled by runningname, so if your document isn't in English, and/or you're unhappy with the default string, you can change it when loading cmsendnotes.

endnotesplit Should the options above not fulfil your needs, you can control more or less all parts of the subdivision of your endnotes section, of the running headers there, and of back references from short notes to full ones, by providing your own sectioning commands in your
NB document. If you wish to use noteref back references in this context, you must first set the biblatex-chicago option endnotesplit to true, no matter which of the two possible implementation methods you choose. With the standard endnotes package you would then use the command \addtoendnotes, while with cmsendnotes it involves variants of the endnote-printing command \theendnotesbypart. (Please note, first, that \addtoendnotes still works with cmsendnotes, in case that's useful to you; and second, that with any other endnote implementation, you'll have to consult its documentation to see if there's a compatible means of dividing the notes.)

Handcrafting w/ cmsendnotes

\theendnotes- bypart[]

\theendnotes- bypart*[]

nosubheader

First, I introduce the methods provided by cmsendnotes. The command \theendnotes-bypart has three variants. The first, \theendnotesbypart*, simply suppresses the printing of the headername, so it works more or less like setting noheader in the options to cmsendnotes. The other two involve an optional argument, in square brackets, containing an individual section number, which prints the endnotes from that section. This command never prints the general endnote section header (as controlled by the headername option), but it will print the individual section's subheader, as controlled by the subheadername option. To turn that printing off you can either use the starred version of the command, i.e., \theendnotesbypart*[], or you can set the cmsendnotes option nosubheader to true. A sequence of commands, each with one section of the document inside square brackets, will give you a complete endnotes section wherever you decide to place it, while the starred forms or nosubheader option allow you to create your own subheaders before each subsection.

NB First, please note that what you need to place inside the square brackets is the *number that forms part of the name of the .ent file in your working directory.* In other words, it's the number *before* any manipulations by the cmsendnotes package remove zeros from it. Depending on the setting of your split option your commands may look like:

```
Chapters                        Sections
\theendnotesbypart*[0]          \theendnotesbypart*[0.0] <-- "introname"
\theendnotesbypart*[1]          \theendnotesbypart*[1.1]
\theendnotesbypart*[2] etc.     \theendnotesbypart*[1.2] etc.
```

When in doubt, have a look in your working directory for the .ent files produced for your document, and use the numbers from there. (If, for some reason, you decide to split by part, you'll probably have roman numerals there, for example, apart from the zero.)

Sectioning The next step is to provide some sort of sectioning command for the subheaders and for the noteref back references. This is slightly complicated, but works the same whether you're using cmsendnotes or endnotes. The basic principle is that the main name of the section appears in the endnotes section, while the optional name provided for the table of contents [toc] appears in the noteref:

```
\subsubsection[chap. 1]{Chapter 1} --> Chapter 1 ... (see chap. 1, n.1)
```

You'll notice that the sectioning command isn't starred, as only unstarred commands provide the optional [toc] argument. (The \addcontentsline command can also be used with starred forms, but keeping the [toc] argument out of the actual table of contents remains an issue, so please read on.) The unusual form of the [toc] argument would merely pollute any table of contents you want to provide, and the actual header in your endnotes section shouldn't have a number in it provided by the standard LaTeX methods, so you'll have to pick a section type that falls underneath the thresholds of the LaTeX counters tocdepth and secnumdepth. By default, in the standard book and report classes,

\subsubsection works for this, while in the article class you may need \paragraph. (You could, of course, also change the counters, should you wish.)

So, let's say you want to subdivide your endnotes section with subheaders containing both the chapter number and the chapter title, as envisaged by the *Manual* (14.46). Your endnotes section might start like this:

```
\section*{Notes}
\subsubsection[intro.]{Introduction: The History of the Problem}
\theendnotesbypart*[0]
\subsubsection[chap. 1]{Chapter 1: Renewing the Question}
\theendnotesbypart*[1]   (etc.)
```

There remain a couple of formatting issues with this code. The endnotes package points out that the first endnote after such a sectioning command won't be indented properly, so it and cmsendnotes use \mbox{}\par\vskip-\baselineskip after sectioning commands to prevent this. Additionally, both packages provide code for running headers using \@mkboth, so if you use such headers you can either do the same inside \makeatletter and \makeatother commands or just use \markboth. Taking all of this into account gives code looking something like this, perhaps:

```
\section*{Notes}
\subsubsection[intro.]{Introduction: The History of the Problem
\markboth{NOTES TO INTRODUCTION}%
    {NOTES TO INTRODUCTION}}%
\mbox{}\par\vskip-\baselineskip
\theendnotesbypart*[0]
\subsubsection[chap. 1]{Chapter 1: Renewing the Question
\markboth{NOTES TO CHAPTER 1}%
    {NOTES TO CHAPTER 1}}%
\mbox{}\par\vskip-\baselineskip
\theendnotesbypart*[1]   (etc.)
```

One of the, perhaps minor, advantages of using the cmsendnotes commands for this is that they will at least all typically be grouped together in one place in your document, rather than scattered throughout, as when you use endnotes' \addtoendnotes command, to which we now turn.

Handcrafting w/ endnotes

\addtoendnotes

To use the endnotes package with its main command \theendnotes to produce a subdivided endnotes section, you must first remember to set the biblatex-chicago option endnotesplit to true, that is, assuming you want to provide noteref back references. For splitting the endnotes, you need the \addtoendnotes command, which you have to place in your document yourself. Ordinarily, you'll need one such command for each relevant division of your text, placed just after the sectioning command itself, so that any endnotes that occur in the section will appear grouped underneath the heading you provide. At the next section, another such command starts a new subsection of endnotes.

To provide the same endnotes section divided by chapter that we've already discussed above, your commands will look something like this:

```
\chapter*{Introduction: The History of the Problem}
\addcontentsline{toc}{chapter}{Introduction}
\addtoendnotes{%
  \protect\subsubsection[intro.]{Introduction: The History of the%
  Problem%
   \protect\markboth{NOTES TO INTRODUCTION}%
   {NOTES TO INTRODUCTION}}%
   \mbox{}\par\vskip-\baselineskip} …
\chapter{Renewing the Question}
\addtoendnotes{%
  \protect\subsubsection[chap. 1]{Chapter 1: Renewing the Question%
    \protect\markboth{NOTES TO CHAPTER 1}%
```

```
{NOTES TO CHAPTER 1}}%
 \mbox{}\par\vskip-\baselineskip} … (etc.)
```

The commands you use are the same as with \theendnotesbypart[], but in this context both the sectioning command *and* the command for running headers need to be \protected. I would also recommend redefining \enoteheading in your preamble, as the default definition produces too much extra vertical space before the first subheading. Something like \def\enoteheading{\section*{Notes}} will do. Finally, remember that you can use cmsendnotes (without a split option) instead of endnotes, if the extra hyperref functionality is important to you. The command sequence above will continue to work in the same way.

There are tradeoffs for both systems. With endnotes, at least the single \theendnotes command keeps things simple, but you still have to keep track of which sections have endnotes in them, else spurious subheaders will appear. Rooting around in your working directory to make sure you've printed all the .ent files is annoying, but at least those represented there will be those which contain endnotes in the first place. Both methods are, I think it's fair to say, a fair amount of labor, but they do give you complete control over how your endnotes section looks, and over how noteref back references within it look. As with all new functionality, noteref and cmsendnotes may well contain bugs, so if you find any please let me know, but do please also send along a minimum working example so I have a chance to identify what's a bug in the code and what's resulted from inadequate documentation.

4.5 General Usage Hints

4.5.1 Loading the Style

With the addition of the author-date styles to the package, I have provided three keys for choosing which style to load, notes, authordate, and authordate-trad, one of which you put in the options to the \usepackage command. The default way of loading the notes + bibliography style has therefore slightly changed. With early versions of biblatex-chicago-notes, the standard way of loading the package was via a call to biblatex, e.g.:

```
\usepackage[style=chicago-notes,strict,backend=bibtex8,%
  babel=other,bibencoding=inputenc]{biblatex}
```

Now, the default way to load the style, and one that will in the vast majority of standard cases produce the same results as the old invocation, will look like this:

```
\usepackage[notes,strict,backend=biber,autolang=other,%
  bibencoding=inputenc]{biblatex-chicago}
```

(In point of fact, the previous biblatex-chicago loading method without the notes option will still work, but only because I've made the notes & bibliography style the default if no style is explicitly requested.) If you read through biblatex-chicago.sty, you'll see that it sets a number of biblatex options aimed at following the Chicago specification, as well as setting a few formatting variables intended as reasonable defaults (see section 4.4.1, above). Some parts of this specification, however, are plainly more "suggested" than "required," and indeed many publishers, while adopting the main skeleton of the Chicago style in citations, nonetheless maintain their own house styles to which the defaults I have provided do not conform.

If you only need to change one or two parameters, this can easily be done by putting different options in the call to biblatex-chicago or redefining other formatting variables in the preamble, thereby overriding the package defaults. If, however, you wish more substantially to alter the output of the package, perhaps to use it as a base for constructing another style altogether, then you may want to revert to the old style of invocation above. You'll lose all the definitions in biblatex-chicago.sty, including those to which I've already alluded and also the code that sets the note number in-line rather than superscript in endnotes or footnotes, the URL line-breaking code, and the Chicago-specific number- and date-range compression code. You'll need to load the required packages xstring and

nameref yourself, as biblatex doesn't do it for you. Also, you'll lose the code that calls cms-american.lbx, which means that you'll lose all the Chicago-specific bibstrings I've defined unless you provide, in your preamble, a \DeclareLanguageMapping command adapted for your setup, on which see section 7 below and also §§ 4.9.1 and 4.11.8 in biblatex.pdf.

What you *will not* lose is the ability to call the package options annotation, strict, short, and noibid (section 4.4.3, above), in case these continue to be useful to you when constructing your own modifications. There's very little code, therefore, actually in biblatex-chicago.sty, but I hope that even this minimal separation will make the package somewhat more adaptable. Any suggestions on this score are, of course, welcome.

4.5.2 Other Hints

One useful rule, when you are having difficulty creating a .bib entry, is to ask yourself whether all the information you are providing is strictly necessary. The Chicago specification is a very full one, but the *Manual* is actually, in many circumstances, fairly relaxed about how much of the data from a work's title page you need to fit into a reference. Authors of introductions and afterwords, multiple publishers in different countries, the real names of authors more commonly known under pseudonyms, all of these are candidates for exclusion if you aren't making specific reference to them, and if you judge that their inclusion won't be of particular interest to your readers. Of course, any data that may be of such interest, and especially any needed to identify and track down a reference, has to be present, but sometimes it pays to step back and reevaluate how much information you're providing. I've tried to make biblatex-chicago-notes robust enough to handle the most complex, data-rich citations, but there may be instances where you can save yourself some typing by keeping it simple.

Scot Becker pointed out to me that the inverse problem not only exists but may well become increasingly common, to wit, .bib database entries generated by bibliographic managers which helpfully provide as much information as is available, including fields that users may well wish not to have printed (ISBN, URL, DOI, pagetotal, inter alia). The standard biblatex styles contain a series of options, detailed in biblatex.pdf § 3.1.2.2, for controlling the printing of some of these fields, and I have implemented the ones that are relevant to biblatex-chicago, along with a couple that Scot requested and that may be of more general usefulness. There is also a general option to excise with one command all the fields under consideration – please see section 4.4.2 above.

If you are having problems with the interaction of punctuation and quotation marks in notes or bibliography, first please check that you've used \mkbibquote in the relevant part of your .bib file. If you are still getting errors, please let me know, as it may well be a bug.

For the biblatex-chicago-notes style, I have fully adopted biblatex's system for providing punctuation at the end of entries. Several users noted insufficiencies in previous releases of biblatex-chicago, sometimes related to the semicolon between multiple citations, sometimes to ineradicable periods after long notes, bugs that were byproducts of my attempt to fix other end-of-entry errors. One of the side effects of this older code was (wrongly) to put a period after a long note produced, e.g., by a command like \footnote{\headlessfullcite}, whereas only the "foot" cite commands (including \autocite in the default biblatex-chicago-notes set up) should do so. If you came to rely on this side effect, please note now that you'll have to put the period in yourself when explicitly calling \footnote, like so: \footnote{\headlessfullcite{key}.}

When you use abbreviations at the ends of fields in your .bib file (e.g., "n.d." or "Inc.,") biblatex-chicago-notes should deal automatically with adding (or suppressing) appropriate punctuation after the final dot. This includes retaining periods after such dots when a closing parenthesis intervenes, as in (n.d.). Merely entering the abbreviation without informing biblatex that the final dot is a dot and not a period should always work, though you do have to provide manual formatting in those rare cases when you need a comma after the author's initials in a bibliography, usually in a misc entry (see house:papers). If you find you need to provide such formatting elsewhere, please let me know.

Finally, allow me to re-emphasize that, in its current form, the notes & bibliography style *requires* the use of biber as your backend — variants of BIBTEX simply cannot produce accurate output anymore, given how many features now depend on the more modern backend.

5 The Specification: Author-Date

The biblatex-chicago package contains two different author-date styles. The first, biblatex-chicago-authordate, implements the specifications of the 17th edition of the *Chicago Manual of Style*. Numbers in parentheses refer to sections of the *Manual*, though many of these references will in fact be to the chapter on the notes & bibliography style (chapter 14), which chapter is, by design, considerably more detailed than that devoted to the author-date style, and which biblatex-chicago-authordate always modifies according to the guidelines in chapter 15. The second author-date style, biblatex-chicago-authordate-trad, implements the same specification but with a markedly different style of title presentation, including sentence-style capitalization and the absence of any quotation marks around the (plain-text) titles of article or incollection entries, *inter alia*. The trad style is so named because older versions of the *Manual*, up to and including the 15th edition, recommended this plainer style for author-date titles, and the 17th edition itself suggests the possibility, when needed, of retaining such title presentation in combination with its own recommendations for other parts of the reference apparatus (15.38). In practice, the differences between the two styles necessitate separate discussions of the title field and one extra package option (headline), and that's about it.

Generally, then, the following documentation covers both Chicago author-date styles, and attempts to explain all the parts of the specification that might be considered somehow "non standard," at least with respect to the styles included with biblatex itself. In the section on entry fields I admit I have also duplicated a lot of the information in biblatex.pdf, which I hope won't badly annoy expert users of the system. As usual, headings in green indicate either material new to this release or old material that has undergone significant revision. The file dates-test.bib contains many examples from the *Manual* which, when processed using biblatex-chicago-authordate, should produce the same output as you see in the *Manual* itself, or at least compliant output, where the specifications are vague or open to interpretation, a state of affairs which does sometimes occur. If you are using biblatex-chicago-authordate-trad the same basically holds, but you'd have to keep one eye on the 15th edition of the *Manual* (chap. 17) for the titles. I have provided cms-dates-sample.pdf and cms-trad-sample.pdf, which show how my system processes dates-test.bib, and I have also included the reference keys from the latter file below in parentheses.

New in this release

5.1 Entry Types

The complete list of entry types currently available in authordate and authordate-trad, minus the odd biblatex alias, is as follows: **article**, **artwork**, **audio**, **book**, **bookinbook**, **booklet**, **collection**, **customc**, **dataset**, **image**, **inbook**, **incollection**, **inproceedings**, **inreference**, **jurisdiction**, **legal**, **legislation**, **letter**, **manual**, **misc**, **music**, **mvbook**, **mvcollection**, **mvproceedings**, **mvreference**, **online** (with its alias **www**), **patent**, **performance**, **periodical**, **proceedings**, **reference**, **report** (with its alias **techreport**), **review**, **standard**, **suppbook**, **suppcollection**, **suppperiodical**, **thesis** (with its aliases **mastersthesis** and **phdthesis**), **unpublished**, and **video**.

What follows is an attempt to specify all the differences between these types and the standard provided by biblatex. If an entry type isn't discussed here, then it is safe to assume that it works as it does in the standard styles. In general, I have attempted not to discuss specific entry fields here, unless such a field is crucial to the overall operation of a given entry type. As a general and important rule, most entry types require very few fields when you use biblatex-chicago-authordate, so it seemed to me better to gather information pertaining to fields in the next section.

article The *Chicago Manual of Style* (14.164) recognizes three different sorts of periodical publication, "journals," "magazines," and "newspapers." The first (14.166) is "a scholarly or pro-

fessional periodical available mainly by subscription," while the second refers to "weekly or monthly (or sometimes daily)" publications that are "available in individual issues at libraries or bookstores or newsstands or offered online, with or without a subscription." "Magazines" will tend to be "more accessible to general readers," and typically won't have a volume number. The following paragraphs detail how to construct your .bib entries for all these sorts of periodical publication.

For articles in "journals" you can simply use the traditional BIBTEX — and indeed biblatex — article entry type, which will work as expected and set off the page numbers with a colon in the list of references, as required by the *Manual*. If, however, you wish to cite a "magazine" or a "newspaper", then you need to add an entrysubtype field containing the exact string magazine or, now, its synonym newspaper. The main formatting differences between a magazine/newspaper and a plain article are that time specifications (month, day, season) aren't placed within parentheses, and that page numbers are set off by a comma rather than a colon. Otherwise, the two sorts of reference have much in common. (For article, see *Manual* 14.168–87, 15.9, 15.46–49; batson, beattie:crime, chu:panda, connell:chronic, conway:evolution, friedman:learning, garaud:gatine, garrett, hlatky:hrt, kern, lewis, loften:hamlet, loomis:structure, rozner:liberation, schneider:mittelpleistozaene, terborgh:preservation, wall:radio, warr:ellison, white:callimachus. For entrysubtype magazine, cf. 14.171, 14.188–200, 15.49; assocpress:gun, lakeforester:pushcarts, morgenson:market, reaves:rosen, stenger:privacy.)

The *Manual* suggests that, no matter which citation style you are using, it is "usually sufficient to cite newspaper and magazine articles entirely within the text" (15.49). This involves giving the title of the journal and the full date of publication in a parenthetical reference, including any other information in the main text (14.198), thereby obviating the need to present such an entry in the list of references. To utilize this method in the author-date styles, in addition to a magazine entrysubtype, you'll need to place cms-date=full into the options field, including skipbib there as well to stop the entry printing in the list of references. If the entry only contains a date and journaltitle that's enough, but if it's a fuller entry also containing an author then you'll also need useauthor=false in the options field. Other surplus fields will be ignored. (See osborne:poison.)

If you are familiar with the notes & bibliography style, you'll know that the *Manual* treats reviews (of books, plays, performances, etc.) as a sort of recognizable subset of "journals," "magazines," and "newspapers," distinguished mainly by the way one formats the title of the review itself. The key rule is this: if a review has a separate, non-generic title (gibbard; osborne:poison) in addition to something that reads like "review of …," then you need an article entry, with or without the magazine entrysubtype, depending on the sort of publication containing the review. If the only title is the generic "review of …," for example, then you'll need the review entry type, with or without this same entrysubtype toggle using magazine. On review entries, see below.

In the case of a review with a specific as well as a generic title, the former goes in the title field, and the latter in the titleaddon field. Standard biblatex intends this field for use with additions to titles that may need to be formatted differently from the titles themselves, and biblatex-chicago-authordate uses it in just this way, with the additional wrinkle that it can, if needed, replace the title entirely, and this in, effectively, any entry type, providing a fairly powerful, if somewhat complicated, tool for getting biblatex to do what you want. Here, however, if all you need is a generic title like "review of …," then you want to switch to the review type, where you can simply use the title field for it.

Biblatex-chicago also, at the behest of Bertold Schweitzer, supports the relatedtype reviewof, which allows you to use the related mechanism to provide information about the work being reviewed. In particular, it relieves you of the need to construct titleaddon or title fields like: review of \mkbibemph{Book Title} by Author Name, as the related entry's title automatically provides the titleaddon in the article type and the title in the review type, with the related mechanism providing the connecting string. This may be particularly helpful if you need to cite multiple reviews of the same work; please see section 5.2.1 for further details.

No less than ten more things need explication under this heading. First, since the *Manual* specifies that what goes into the titleaddon field of article entries stays unformatted —

no italics, no quotation marks — this plain style is the default for such text, which means that you'll have to format any titles within titleaddon yourself, e.g., with \mkbibemph{}. Second, the *Manual* specifies a similar plain style for the titles of other sorts of material found in "magazines" and "newspapers," e.g., obituaries, letters to the editor, interviews, the names of regular columns, and the like. References may contain both the title of an individual article and the name of the regular column, in which case the former should go, as usual, in a title field, and the latter in titleaddon. As with reviews proper, if there is only the generic title, then you want the review entry type. (See 14.191, 14.195–96; morgenson:market, reaves:rosen.)

Third, the *Manual* suggests that "unsigned newspaper articles or features are best dealt with in text ..." (14.199). As with newspaper or magazine articles in general, you can place cmsdate=full and skipbib into the options field to produce an augmented in-text citation whilst keeping this material out of the reference list. If you do use the reference list, then the standard shorter citation will be sufficient, and in both cases the name of the periodical (in the journaltitle field) will be used in place of the missing author. Just to clarify: in article or review entries, entrysubtype magazine, a missing author field results in the name of the periodical (in the journaltitle field) being used as the missing author. Without an entrysubtype, and assuming that no name whatsoever can be found to put at the head of the entry, the title will be used, not the journaltitle, or so I interpret the *Manual* (14.168). The default sorting scheme in biblatex-chicago-authordate considers the journaltitle before the title, so if the latter heads an entry you'll need a sortkey, just as you will if you retain the definite or indefinite article at the beginning of the journaltitle in author-less entries with an entrysubtype. If you want to abbreviate the journaltitle for use in citations, but give the full name in the list of references, then the shortjournal field is the place for it. A shortened title should go, as usual, in shorttitle. (See section 5.4.1, below; lakeforester:pushcarts, nyt:trevorobit, unsigned:ranke.)

Fourth, Bertold Schweitzer has pointed out, following the *Manual* (14.183), that while an issuetitle often has an editor, it is not too unusual for a title to have, e.g., an editor and/or a translator. In order to allow as many permutations as possible on this theme, I have brought the article entry type into line with most of the other types in allowing the use of the namea and nameb fields in order to associate an editor or a translator specifically with the title. The editor and translator fields, in strict homology with other entry types, are associated with the issuetitle if one is present, and with the title otherwise. The usual string concatenation rules still apply — cf. editor and editortype in section 5.2, below.

Fifth, in certain fields, just beginning your data with a lowercase letter activates the mechanism for capitalizing that letter depending on its context within a reference list entry. This is less important in the author-date styles, where this information only turns up in the reference list and not in citations, but you can consult **\autocap** in section 5.3.1 below for all the details. Both the titleaddon and note fields are among those treating their data this way, and since both appear regularly in article entries, I thought the problem merited a preliminary mention here.

Sixth, if you need to cite an entire issue of any sort of periodical, rather than one article in an issue, then the periodical entry type, once again with or without the magazine toggle in entrysubtype, is what you'll need. (You can also use the article type, placing what would normally be the issuetitle in the title field and retaining the usual journaltitle field, but this arrangement isn't compatible with standard biblatex.) The note field is where you place something like "special issue" (with the small "s" enabling the automatic capitalization routines), whether you are citing one article or the whole issue (conley:fifthgrade, good:wholeissue). Indeed, this is a somewhat specialized use of note, and if you have other sorts of information you need to include in an article or periodical entry, then you shouldn't put it in the note field, but rather in titleaddon or perhaps addendum (brown:bremer).

Seventh, I would suggest that if you wish to cite certain kinds of television or radio broadcast, most notably interviews but perhaps also news segments or other "journalistic" material, then the article type, entrysubtype magazine is the place for it. The name of the program as a whole would go in journaltitle, with the name of the episode in title. The network's name goes into the usera field. Of course, if the piece you are citing has

only a generic name (an interview, for example), then the review type would be the best place for it (8.189, 14.213; see bundy:macneil for an example of how this all might look in a .bib file.) Other sorts of broadcast, usually accessible through commercial recordings, would need one of the audiovisual entry types, probably audio (danforth:podcast) or video (friends:leia), while recordings from archives fit best either into online or into misc entries with an entrysubtype (coolidge:speech, roosevelt:speech).

Eighth, the *Manual* (14.208, 15.51) specifies that blogs and other, similar online material should be presented like articles, with magazine entrysubtype (ellis:blog), and needn't appear in a reference list at all, if you'd prefer to provide relevant details in the text. I've attempted, however, to make biblatex-chicago-authordate as useful as possible when managing references to such sources, so I'll outline these facilities here. The title of the specific entry goes in title, the general title of the blog goes in journaltitle, and the word "blog" in the location field (though you could just use special formatting in the journaltitle field itself, which may sometimes be necessary). The 17th edition specifies that "blogs that are part of a larger publication should include the name of that publication." This usually involves a newspaper or magazine which also publishes various blogs on its website, and it means that such entries need a more general title than the journaltitle. It's not standard biblatex or anything, but you can now put such information in maintitle (with mainsubtitle and maintitleaddon, if needed), but only in article and review entries with a magazine entrysubtype (amlen:hoot).

The *Manual* (15.51) is even more emphatic about whole blogs (rather than individual posts) and comments on blogs not appearing in reference lists, but I've kept as many options open as possible, including fairly simple ways you can provide all the information needed in text citations alone. To cite a whole blog, you'll need the periodical entry type, with a title instead of a journaltitle, along with a (possible) maintitle (amlen:wordplay). Comments on blogs, with generic titles like "comment on" or "reply to," need a review entry with the same entrysubtype (viv:amlen). Such comments make particular use of the eventdate and nameaddon fields, and also of specialized **customc** entries for adding comments to in-text citations. Please see the documentation of **customc**, **periodical**, and **review**, the relatedtype commenton in section 5.2.1, and the general discussion of online sources in the **online** documentation.

Ninth, the special biblatex field shortjournal allows you to present shortened journaltitles in article, review, and periodical entries, as well as facilitating the creation of lists of journal abbreviations in the manner of a shorthand list. Please see the documentation of **shortjournal** in section 5.2 for all the details on how this works.

Finally, the 17th edition (14.191) specifies that, for news sites carrying "stories as they unfold, it may be appropriate to include a time stamp for an article that includes one." You can provide this by using the standard biblatex time stamp format inside the date field, e.g., 2008-07-01T10:18:00. Since the *Manual* prefers the standard time zone initialisms, a separate timezone field would be required if you want to provide one.

If you're still with me, allow me to recommend that you browse through dates-test.bib to get a feel for just how many of the *Manual*'s complexities the article, periodical, and review types attempt to address. It may be that in future releases of biblatex-chicago I'll be able to simplify these procedures somewhat, but with any luck the vast majority of sources won't require knowledge of these onerous details.

artwork Arne Kjell Vikhagen pointed out to me that none of the standard entry types were straightforwardly adaptable when referring to visual artworks. It's unclear whether the *Manual* (14.235) believes it necessary to include them in the reference apparatus at all, but it's easy to conceive of contexts in which a list of artworks studied might be desirable, and biblatex includes entry types for just this purpose, though the standard styles leave them undefined. Biblatex-chicago defines both artwork and image, which are in fact now clones of each other, so you can use either of them indifferently, the distinction existing only for historical reasons.

As one might expect, the artist goes in author and the name of the work in title. The type field is intended for the medium — e.g., oil on canvas, charcoal on paper — and the

version field might contain the state of an etching. You can place the dimensions of the work in note, and the current location in organization, institution, and/or location, in ascending order of generality. The type field, as in several other entry types, uses biblatex's automatic capitalization routines, so if the first word only needs a capital letter at the beginning of a sentence, use lowercase in the .bib file and let biblatex handle it for you. (See *Manual* 3.22, 8.198; leo:madonna, bedford:photo.)

The 17th edition of the *Manual* has included new information in some of its examples, so I have added 4 new fields to the driver. Alongside the usual date for the creation of a work, you may also want to include the printing date of a particular exemplar of a photograph or a print. The system I have designed uses the *earlier* of the date and the origdate to be the date of creation, and the *later* to be the printing date. The style will automatically prefix the printing date with the localized \bibstring printed, so if that's the wrong string entirely then you can define userd any way you like to change it. If only *one* of those two dates is available, it will always serve as a creation date. Any date specification provided will always appear in full somewhere in the reference list entry, though sometimes that could be the plain year at the head of the entry. This system, which is unlike other entry types, helps to avoid ambiguities in some situations.

One of the *Manual's* examples is of a photograph published in a periodical, and information about this publication appears late in the entry, after the type. I have included the howpublished field so that you can give information about the periodical (meaning that you'll have to format the title yourself with \mkbibemph), and the eventdate field for you to provide the date of publication (mccurry:afghangirl).

Depending on the presence or absence of these three date fields, and also on how you've set the cmsdate option, any of the three can appear in citations and at the head of reference list entries, allowing you to order entries by creation date, printing date, or publication date. See the documentation of **date** on page 120, below, for all the complicated details. Please note, when choosing your date presentation, that these new fields ostensibly replace most of the possible uses of the pubstate field in artwork entries, though this field continues to function here more or less as before, should you still require it.

As a final complication, the *Manual* (8.198) says that "the names of works of antiquity ... are usually set in roman." If you should need to include such a work in the reference apparatus, you can either define an entrysubtype for an artwork entry — anything will do — or you could try the misc entry type with an entrysubtype. Assuming the complicated date handling I've just outlined isn't required for such a work, in this instance the other fields in a misc entry function pretty much as in artwork.

audio Following the request of Johan Nordstrom, I have included three entry types, all undefined by the standard styles, designed to allow users to present audiovisual sources in accordance with the Chicago specifications. The *Manual's* presentation of such sources (14.261–68, 15.57), though admirably brief, seems to me somewhat inconsistent, though perhaps I'm merely unable to spot the important regularities. The proliferation of online sources has made the task yet more complex, requiring the inclusion of the **article**, the **online**, and even the **misc** entry types, which see, under the audiovisual rubric. I shall attempt to delineate the main differences here, and though there are likely to be occasions when your choice of entry type is not obvious, at the very least biblatex-chicago should help you maintain consistency.

For users of the author-date styles, the 17th edition of the *Manual* continues to emphasize the need to provide dating information for audiovisual materials (14.263), meaning that nearly all such entries will have some such information and will therefore fit better, stylistically, with other author-date references. In particular, it continues to recommend that "the date of the original recording should be privileged in the citation" (15.57). Guidance for supplying dates for this class of material will be found below under the different entry types in use, though it will also be worthwhile to look at the documentation of date, eventdate, origdate, and urldate, in section 5.2, below. The *Manual* continues to suggest, also, that "it is often more appropriate to list such materials in running text and group them in a separate section or discography."

The **music** type is intended for all musical recordings that do not have a video component. This means, for example, digital media (whether on CD or hard drive), vinyl records, and tapes. The **video** type includes most visual media, whether it be films, TV shows, tapes and DVDs of the preceding or of any sort of performance (including music), or online multimedia. The *Manual's* treatment (14.267) of the latter suggests that online video excerpts, short pieces, and interviews should generally use the **online** type (horowitz:youtube, pollan:plant) or the **article** type (harwood:biden, kessler:nyt), depending on whether the pieces originate from an identifiably "journalistic" outlet. The **audio** type, our current concern, fills gaps in the others, and presents its sources in a more "book-like" manner. Published musical scores need this type — unpublished ones would use misc with an entrysubtype (shapey:partita) — as do podcasts and such favorite educational formats as the slideshow and the filmstrip (danforth:podcast, greek:filmstrip, schubert:muellerin, verdi:corsaro). The *Manual* (14.264) sometimes uses a similar format for audio books (twain:audio), though, depending on the sorts of publication facts you wish to present, this sort of material may fall under music (auden:reading). Dated audio recordings that are part of an archive, online or no, may be presented either in an **online** or in a **misc** entry with an entrysubtype, the difference mainly being in just how closely associated the date will be with the title (coolidge:speech, roosevelt:speech). Actual radio broadcasts (as opposed to podcasts) pose something of a conundrum. Interviews and other sorts of "journalistic" material fit well into article or review entries (14.213), but other sorts of broadcast are not well represented in the *Manual's* examples (8.189), and what little there is suggests that, counter-intuitively, the video type is the best fit, as it is well equipped to present broadcasts of any sort.

Once you've accepted the analogy of composer to author, constructing an audio entry should be fairly straightforward, since many of the fields function just as they do in book or inbook entries. Indeed, please note that I compare it to both these other types as, in common with the other audiovisual types, audio has to do double duty as an analogue for both books and collections, so while there will normally be an author, a title, a publisher, a date, and a location, there may also be a booktitle and/or a maintitle — see schubert:muellerin for an entry that uses all three in citing one song from a cycle. (As with the music and video types, you can cite an individual piece separate from any large collection by using the title field and by defining an entrysubtype, which will stop biblatex-chicago italicizing your title in the absence of a booktitle.) If the medium in question needs specifying, the type field is the place for it. Please note, also, that while the titleaddon field can still specify creative or editorial functions for which biblatex-chicago provides no automated, localized handling, you can also now provide the string you need in an editor[abc]type field, e.g., "libretto by" (verdi:corsaro).

For podcasts, newly covered by the 17th edition (14.267), the audio type provides the nearest analogue I could find, and in general most of the data should fit comfortably into the fields already discussed above, the episode name in title and the name of the podcast in booktitle, for starters. Two details, however, need mentioning: first, the note field as the place to specify that it is a podcast, and the eventdate field for the date of publication of the specific episode (title) cited, which appears in close association with that title. Indeed, the eventdate field helps biblatex-chicago know that the audio entry is a podcast episode, and helps it construct the entry appropriately (danforth:podcast).

book   This is the standard biblatex and BIBTEX entry type, but the package can automatically provide abbreviated references in the reference list when you use a crossref or an xref field. The functionality is not enabled by default, but you can enable it in the preamble or in the options field using the booklongxref option. Please see **crossref** in section 5.2 and booklongxref in section 5.4.2, below. Also, cf. harley:ancient:cart, harley:cartography, and harley:hoc for how this might look. The book type is useful also to present multimedia app content, the added fields version and type providing information about the app's version and about the system on which it runs (14.268, 15.57; angry:birds).

bookinbook   This type provides the means of referring to parts of books that are considered, in other contexts, themselves to be books, rather than chapters, essays, or articles. Such an entry can have a title and a maintitle, but it can also contain a booktitle, all three of which will

be italicized in the reference matter. In general usage it is, therefore, rather like the traditional inbook type, only with its title in italics rather than in quotation marks. As with the book type, you can enable automatically abbreviated references in the reference list, though this isn't the default. Please see **crossref** in section 5.2 and booklongxref in section 5.4.1, below. (Cf. *Manual* 14.109, 14.122, 14.124; bernhard:boris, bernhard:ritter, and bernhard:themacher for the abbreviating functionality; also euripides:orestes [treated differently in 14.122 and 14.124], plato:republic:gr.)

booklet This is the first of two entry types — the other being manual, on which see below — which are traditional in BIBTEX styles, but which the *Manual* (14.220) suggests may well be treated basically as books. In the interests of backward compatibility, biblatex-chicago-authordate will so format such an entry, which uses the howpublished field instead of a standard publisher, though of course if you do decide just to use a book entry then any information you might have given in a howpublished field should instead go in publisher. (See clark:mesopot.)

collection This is the standard biblatex entry type, but the package can provide automatically abbreviated references in the reference list when you use a crossref or an xref field. The functionality is not enabled by default, but you can enable it in the preamble or in the options field using the booklongxref option. Please see **crossref** in section 5.2 and booklongxref in section 5.4.2, below. See harley:ancient:cart, harley:cartography, and harley:hoc for how this might look.

customc This entry type allows you to include alphabetized cross-references to other, separate entries in the bibliography, particularly to other names or pseudonyms, as recommended by the *Manual*. (This is different from the usual crossref, xref, userf, and related mechanisms, all primarily designed to include cross-references to other works. Cf. 14.81–82, 15.35). I should add immediately that, as I read the specification, the alphabetized cross-references provided by customc are particularly encouraged, bordering on required, when a reference list "includes two or more works published by the same author but under different pseudonyms." The following entries in dates-test.bib show one way of addressing this: creasey:ashe:blast, creasey:york:death, creasey:morton:hide, ashe:creasey, york:creasey and morton:creasey. In these latter cases, you would need merely to place the pseudonym in the author field, and the author's real name, under which his or her works are presented in the bibliography, in the title field. To make sure the cross-reference also appears in the bibliography, you can either manually include the entry key in a \nocite command, or you can put that entry key in the **userc** field in the work's main .bib entry, in which case biblatex-chicago will print the cross-reference if and only if you cite the main work. (Cf. userc, below.)

Under ordinary circumstances, biblatex-chicago will connect the two parts of the cross-reference with the word "*See*" — or its equivalent in the document's language — in italics. If you wish to present it differently, you can put the connecting word(s) into the nameaddon field, formatted as you wish.

Finally, you may need to use this entry type if you wish to include a comment inside the parentheses of a citation, as specified by the *Manual* (15.24). If you have a postnote, then you can manually provide the punctuation and comment there, e.g., \autocite[4; the unrevised trans.]{stendhal:parma}. Without a postnote, you have two solutions. You can enable the postnotepunct option, which allows you simply to type \autocite[; the unrevised trans.]{stendhal:parma}, or you can use a separate customc entry containing just the text of the comment in the title field, entrysubtype classical, and options skipbib. An \autocites command calling both the main text and the comment then does the trick, e.g., \autocites{chicago:manual}{chicago:comment}. Cf. postnotepunct in section 5.4.3, below.

For its 17th edition, the *Manual* has provided a more detailed treatment of online comments, whether on blogs, mailing lists, or social media posts (15.50–52). Such comments "are cited only in the text, in reference to the related post," an arrangement most easily created using customc entries referencing the main post. The new commenton relatedtype in online and review entries attempts to automate this for you, creating a separate

customc entry to be used in an \autocites command together with the comment's own entry. Please see the details of this in the **online** and **review** entry types, below, and in the commenton docs in section 5.2.1. Cf. also ac:comment, diaz:surprise, ellis:blog, licis:diazcomment, and the associated automatically-created entries ellis:blog-customc and diaz:surprise-customc.

dataset   This entry type, new in biblatex 3.13, allows you to cite scientific databases, for which the *Manual* (14.257) presents some rather specific, if brief, instructions. To construct your entry, you can put the name of the database into author, a "descriptive phrase or record locator" in the title field, and if there's a specific accession number needed beyond the record locator you can put it into the number field, with the type field reserved to help explain what sort of number is involved. The howpublished field can also be used to provide extra descriptive detail about the number, if needed. More generally, a url will locate the database as a whole and a urldate will specify the date you accessed it. If, for some reason, an additional date is relevant, then the date field is available, while the pubstate field will appear before the date in case you need to modify the latter. (See 14.257; genbank:db, nasa:db.) Given that usually the only relevant date in such entries is the access date, which means that they would usually have "n.d." at the head of reference-list entries and in citations, I have thought it sensible to treat dataset entries by default as author-title instead of author-date in citations, and to stop the printing of "n.d." in reference lists just as in misc entries. You can shorten the component parts of the author-title citation using the usual shortauthor, shorthand, and/or shorttitle fields, and you can also restore author-date formatting to these entries by setting the authortitle option to false either in individual entries or in the preamble for all examples of the entry type.

image   This entry type is now a clone of the artwork type, which see. I retain it here for historical reasons (See 3.22, 8.198; bedford:photo.)

inbook   These two standard biblatex types have very nearly identical formatting requirements
incollection   as far as the Chicago specification is concerned, but I have retained both of them for compatibility. Biblatex.pdf (§ 2.1.1) intends the first for "a part of a book which forms a self-contained unit with its own title," while the second would hold "a contribution to a collection which forms a self-contained unit with a distinct author and its own title." The title of both sorts will be placed within quotation marks, and in general you can use either type for most material falling into these categories. I have, in both types, implemented the *Manual's* recommendations for space-saving abbreviations in the reference list when you cite multiple pieces from the same collection. These abbreviations are activated by default when you use the crossref or xref field in incollection entries *and* in inbook entries, because although the *Manual* (15.42) here specifies a "multiauthor book," I believe the distinction between the two is fine enough, and the discussion general enough, to encourage similar treatments. (For more on this mechanism see **crossref** in section 5.2, below, and the option longcrossref in section 5.4.2. Please note that it is also active by default in letter and inproceedings entries.) If the part of a book to which you are referring has had a separate publishing history as a book in its own right, then you may wish to use the bookinbook type, instead, on which see above. (See *Manual* 14.106–9, 15.42; inbook: ashbrook:brain, phibbs:diary, will:cohere; incollection: centinel:letters, contrib:contrib, sirosh:visualcortex; ellet:galena, keating:dearborn, and lippincott:chicago [and the collection entry prairie:state] demonstrate the use of the crossref field with its attendant abbreviations in the list of references.)

NB: The *Manual* suggests that, when referring to a chapter, one use either a chapter number or the inclusive page numbers, not both. In-text citations, of course, require any postnote field to specify if it is a whole chapter to which you are referring.

inproceedings   This entry type works pretty much as in standard biblatex, even more so now that, after a request from Patrick Danilevici, I have newly included the eventdate, eventtitle, eventtitleaddon, and venue fields for specifying where and when the event occurred that produced the proceedings. These four fields are the main difference between it and incollection, along with the lack of an edition field and the possibility that an organization may be cited alongside the publisher, even though the *Manual* doesn't specify the use of any of

these supplementary fields (14.217). Please note, also, that the crossref and xref mechanism for shortening citations of multiple pieces from the same proceedings is operative here, just as it is in incollection and inbook entries. See **crossref** in section 5.2 and the option longcrossref in section 5.4.2, below, for more details.

inreference This entry type is aliased to incollection in the standard styles, but the *Manual's* requirements prompted a thoroughgoing revision. Instructions for the author-date style are not copious, so what follows is my best guess at following the specification (14.232–34).

Please first note that if your reference work can easily or conveniently be presented like a regular author-date book, that is, with an author or editor, a year of publication, and a title, and if you will be citing it by page or section number, then you should almost certainly simply choose the book entry type for your .bib entry. (Cf. mla:style, schellinger:novel, times:guide. The latter was presented as an inreference entry for the notes & bibliography style, but because the book entry type can also present references to alphabetized headings [see below], at least in the list of references, then it seemed better just to choose a book entry for the author-date styles.)

If your source simply doesn't fit the standard author-date template for a book, most especially if it is a "well-known" reference work, then you may need to use the inreference type, the main feature of which is the lista field, which you use to present citations from "alphabetically arranged" works by named article rather than by page number. You should present these article names just as they appear in the work, separated by the keyword "and" if there is more than one, and biblatex-chicago-authordate will provide the appropriate prefatory string (s.v., plural s.vv.), and enclose each in its own set of quotation marks (times:guide). More relevant to the author-date styles is the fact that the postnote field works the same way in inreference entries, the only limitation on this system being that this field, unlike lista, is not a list, and therefore for the formatting to work correctly you can only put one article name in it. In the case of "[w]ell-known reference books, such as major dictionaries and encyclopedias," you are encouraged not to include them in the list of references, so the lista field actually may be of less use than this special formatting of postnote. You may want to look at ency:britannica, where only a (carefully-formatted) shorttitle, a year, and an options field are necessary to allow you to produce in-text citations that look like (*Ency. Brit.*, 15th ed. 1985, s.v. "Article").

If it seems appropriate to include such a work in the list of references, perhaps because the work is not so well known that a short citation will be parseable by your readers, or perhaps because it is an online work, which requires you to provide a urldate (see below), be aware that the contents of the lista field will also be presented there, which may not be what you want. A separate inreference or reference entry might well solve this problem. In a typical inreference entry very few fields are needed, but "if a physical edition is cited, not only the edition number (if not the first) but also the date the volume or set was issued must be specified." In practice, this means a title, date, and possibly an edition field. The author field holds the author of the specific article (in lista), not the author of the title as a whole. This name will be printed in parentheses after the alphabetized entry's title (grove:sibelius).

Finally, all of these rules apply to online reference works, along with a few more. The 17th edition of the *Manual* now allows, "subject to editorial discretion," the alternative treatment of an online reference work which "does not have (and never had) a printed counterpart" (14.206, 14.233). In effect this means that it can be treated more like an online entry than a book, its title therefore in plain roman rather than in italics. You can achieve this in inreference entries by providing an entrysubtype in the entry. Online reference works need not only a url but also, always, a urldate (instead of a date), as these sources are in constant flux. When that flux is of a particularly high frequency, as with Wikipedia, then a time stamp may also be needed. You can provide this in the urldate field
urlstamp=true itself, using the standard biblatex format, e.g., 2008-07-01T10:18:00. It is possible to turn off the printing of the urltime with the new urlstamp option, which is set to true by default, but which can be changed in your preamble for the whole document, for specific entry types, or in the options field of specific entries (wikiped:bibtex, grove:sibelius). In keeping with the rules of the 17th edition, changed since the 16th, any inreference entry

with no date, or with only an access date, as opposed to a revision date or another, more traditional publishing date, will use the `nodate` abbreviation at the head of the entry and in citations (15.44, 15.50).

jurisdiction
legal
legislation

I document these three types in section 6 below, both because they all follow the specifications of the *Bluebook* instead of the *Manual*, and also because they are the only entry types treated identically by the notes & bibliography style and the author-date styles.

letter

This entry type was designed to be used for citing letters, memoranda, or similar texts, but *only* when they appear in a published collection. (Unpublished material of this nature needs a misc entry, for which see below.) The author-date specification (15.43), however, recommends against individual letters appearing in a list of references, suggesting instead that you put the whole published collection in a book entry and use a notice in the text to specify the letter (white:total).

If you absolutely must include individual letters in the list of references, for whatever reason, then please consult the instructions above for the notes & bibliography style in section 4.1, s.v. "letter," which cover all the details. There are a few wrinkles, related to date specifications, that I shall attempt to clarify here. If you look at white:ross:memo and white:russ, you'll see two letters from the same published collection, both written in the same year. You can simply use the origdate field in both of them, because in the absence of a date (or an eventdate) Biber and biblatex will use the origyear as the labelyear, putting it at the head of the entry and in the citation, and also ensuring that the letters a,b,c are appended to disambiguate the two sources. In this case, it works because we are using the xref mechanism to refer to the whole published collection (white:total), so a separate citation of that entry provides the date for the shortened cross-reference included in the list of references, and the letter entry never sees that date at all. (Cf. also the documentation of the date field in section 5.2 below.)

manual

This is the second of two traditional BIBTEX entry types that the *Manual* suggests formatting as books, the other being booklet. As with this latter, I have retained it in biblatex-chicago-authordate for backward compatibility, its main peculiarity being that, in the absence of a named author, the organization producing the manual will be provided both as author and as publisher. (You can give a shortened form of the organization in the shortauthor field for text citations, if needed, or use the shorthand field.) Of course, if you were to use a book entry for such a reference, then you would need to define both author and publisher using the name you here might have put in organization. (See 14.84; chicago:manual, dyna:browser, natrecoff:camera. Cp. also the new **standard** entry type.)

misc

As its name suggests, the misc entry type was designed as a hold-all for citations that didn't quite fit into other categories. In biblatex-chicago, I have somewhat extended its applicability, while retaining its traditional use. Put simply, with no entrysubtype field, a misc entry will retain backward compatibility with the standard styles, so the usual howpublished, version, and type fields are all available for specifying an otherwise unclassifiable text, and the title will be italicized. (The *Manual*, you may wish to note, doesn't give specific instructions on how such citations should be formatted, so when using the Chicago style I would recommend you have recourse to this traditional entry type as sparingly as possible.)

If you do provide an entrysubtype field, the misc type provides a means for citing unpublished letters, memoranda, private contracts, wills, interviews, and the like, making it something of an unpublished analogue to the letter, article, and review entry types (which see). It also works well for presenting online audio pieces, particularly dated ones, like speeches. Typically, such an entry will cite part of an archive, and equally typically the text cited won't have a specific title, but only a generic one, whereas an unpublished entry will ordinarily have a specific author and title, and won't come from a named archive. The misc type with an entrysubtype defined is the least formatted of all those specified by the *Manual*, so titles are in plain text by default. (It is quite possible, though somewhat unusual, for archival material to have a specific title, rather than a generic one. In these cases, you will need to enclose the title inside a `\mkbibquote` command manually. Cf. roosevelt:speech, shapey:partita.)

If you are presenting part of an unpublished archive, then it's worth remembering that, as with the letter type, the *Manual* (15.54) suggests that the list of references will usually contain only the name of the whole archived collection, with more specific information about individual items provided in the text, outside the parentheses. If, on the other hand, "only one item from a collection has been mentioned in text, the entry may begin with the writer's name (if known)." (See 14.211–12, 14.219, 14.221–31; house:papers cites a whole archive, while creel:house, dinkel:agassiz, and spock:interview cite individual pieces.)

As far as constructing your .bib entry goes, you should first know that the absence of any date will not result in the "n.d." abbreviation automatically being provided. Indeed, no date at all will be required for entries referring to entire archival collections. In such entries, you may wish to use the word "classical" as your entrysubtype, which will have no effect on the list of references but will change the look of the in-text citations (house: papers). Instead of any date, the citation will include the title, separated from the author's name by a space, e.g., (House Papers). This same arrangement, happily, allows you easily to cite individual books of the Bible, and also certain other sacred texts (14.238–41; genesis). Please see under entrysubtype in section 5.2 below for all the details of the classical toggle.

As for presenting the date of individual items, the *Manual* (14.224) allows in these entries, as it does in documentation generally "if numerous dates occur" (9.35), for a more streamlined presentation of dates using the day-month-year form, different from the standard American month-day-year. In letter entries you use the origdate field to give the date of individual letters, and it is always presented in the more streamlined form. Here, the same field will do exactly the same thing, though with the added wrinkle that if you'd prefer to use the standard day-month-year form you can, simply by putting the date into the date field instead. (Please choose one only in misc entries with an entrysubtype — in letter entries the date refers to the published collection.) As with the letter type, if the only date present is an origdate, you don't need to set the cmsdate option in your .bib entry to make sure that that year appears at the head of the entry (and in citations) — this happens automatically. (Cf. particularly the documentation in section 5.2 below, s.v. "date", and also the letter type above for some of the date-related complications that can arise, and how you can address them with judicious use of the options, date, and origdate fields.)

As in letter entries, the titles of unpublished letters are of the form Author to Recipient, further information can be given in the titleaddon field, while the origlocation field can hold the place where the letter was written. Interviews or similar pieces will have a different sort of title, but all types will use the note, organization, institution, and location fields (in ascending order of generality) to identify the archive, though the *Manual* specifies (14.227) that well-known depositories don't usually need a city, state or country specified. (The traditional misc fields are all still available, also.)

There are a few more subtleties involved here. Some material (roosevelt:speech) includes a venue for the event recorded in the archive, so I have added the venue field, which prints *before* the date, with the origlocation appearing after it. Somewhat confusingly, in published letters the origlocation itself prints before the date, rather than after, so if the inconsistency between published and unpublished letters bothers you then you could conceivably use venue instead of origlocation for that purpose here. Also, the *Manual* offers several examples of specific location information *within* an archive, some of which appears *before* the main archive name, and some of which appears *after* it. I assume this may depend on the exact nature of the archive itself, but in any case you can try the type or howpublished fields for the first case and the number field for the second.

In all this class of archived material, the *Manual* (14.221) quite specifically requires more consistency within your own work than conformity to some external standard, so it is the former which you should pursue. I hope that biblatex-chicago proves helpful in this regard.

music   This is one of three audiovisual entry types, and is intended primarily to aid in the presentation of musical recordings that do not have a video component, though it can also

include audio books (auden:reading). A DVD or VHS of an opera or other performance, by contrast, should use the **video** type instead, while an online music video will probably need an **online** entry. (Cf. online and video; handel:messiah, horowitz:youtube.) Because biblatex — and BIBTEX before it — were designed primarily for citing book-like objects, some choices needed to be made in assigning the various roles found on the back of a CD to the fields in a typical .bib entry. I have also implemented several bibstrings to help in identifying these roles within entries. The 17th edition of the *Manual* once again revised its recommendations for this type, but fortunately the changes are additive, i.e., you can re-use 16th-edition citations but are encouraged to peruse the following guidelines to see if there's any information you might think of adding to bring your citations more into line with the spec.

These guidelines, in summary form, are:

author = composer, songwriter, or performer(s), depending on whom you wish to emphasize by placing them at the head of the entry.

bookauthor: Somewhat like an author, but it will hold the name associated with a whole album rather than an individual piece, should both be present, and will therefore appear in close association with the booktitle rather than at the head of the entry.

editor, editora, editorb = conductor, director or performer(s). These will ordinarily follow the title of the work, though the usual useauthor and useeditor options can alter the presentation within an entry. Because these are non-standard roles, you will need to identify them using the following:

editortype, editoratype, editorbtype: The most common roles, all associated with specific bibstrings (or their absence), will be conductor, director, producer, and, oddly, none. The last is particularly useful when identifying the group performing a piece, as it usually doesn't need further specifying and this role prevents biblatex from falling back on the default editor bibstring. The 17th edition (15.57) also seems to favor, in some circumstances, using strings to identify individual performers, e.g., "vocalist" or "pianist," so even though there's no \bibstring associated with these types you can now provide them, or anything else you need, in whatever form ("vocalist" or "sung by") suits your citation.

note: This field can also hold contributors, perhaps collaborators or featured artists (holiday:fool, rihanna:umbrella).

title, booktitle, maintitle: As with the other audiovisual types, music serves as an analogue both to books and to collections, so the title will either be, e.g., the album title or a song title, in which latter case the album title would go into booktitle. If you wish to cite a song that, as may happen, isn't part of any larger collection, your entry will in such a case have only a title, which biblatex-chicago would normally interpret as an album title. You can now define an entrysubtype to let it know that the lone title is in fact a song (cf. naraya). The maintitle might be necessary for something like a box set of *Complete Symphonies*.

chapter: The 17th edition seems more keen on having track numbers for individual pieces, whether on a traditional format or on a streaming service. The chapter field is the place for this information, and biblatex-chicago will automatically prepend the localized string track (cf. holiday:fool, rihanna:umbrella).

publisher, series, number: These three closely-associated fields are intended for presenting the catalog information provided by the music publisher. The 17th edition generally only requires the series and number fields (nytrumpet:art), which hold the record label and catalog number, respectively. Alternatively, publisher would function as a synonym for series (holiday:fool), but there may be cases when you need or want to specify a publisher in addition to a label, as perhaps when a single publisher oversees more than one label. You can certainly put all of this information into one of the above fields, but separating it may help make the .bib entry more readable.

pubstate: The pubstate field in music entries mainly has the usual meaning it has in other entry types, for which see the documentation of the field in section 5.2, below. If the field contains reprint, however, this has a special meaning in music entries, where it will transform the origdate from a recording date for an entire album into an original release date for that album, notice of which will be printed towards the end of a reference list entry, always assuming that the origdate hasn't already appeared at the head of the entry and in citations. No reprint \bibstring will be printed, as only the syntax of the reference will have been altered.

date, eventdate, origdate: The 17th edition of the *Manual*, like the 16th, considers music citations without a date to be "generally unacceptable" (14.263), while if there is more than one date "the date of the original recording should be privileged" (15.57). Finding these dates may take some research, but they will basically fall into two types, i.e., the date of the recording or the copyright / publishing date. Recording dates go either in origdate (for complete albums) or eventdate (for individual tracks). The current copyright or publishing date goes in the date field, while the original release date goes in origdate. You may have noticed that the origdate has two slightly different uses — you can tell biblatex-chicago which sort you intend by using the string reprint in the pubstate field, which transforms the origdate from a recording date into an original release date. The style will automatically use the eventdate or the origdate in citations and at the head of the list of references, falling back on a date or even a urldate in their absence. It will also prepend the bibstring recorded to any part of the eventdate that doesn't appear at the head of the list of references or, in the absence of the pubstate mechanism, to the origdate, or indeed to both. You can modify what is printed here using the userd field, which acts as a sort of date type modifier. In music entries, userd will be prepended to an eventdate if there is one, barring that to the origdate, barring that to a urldate, and absent those three to a date. (See floyd:atom, holiday:fool, nytrumpet:art.)

type, howpublished: As in all the audiovisual entry types, the type field holds the medium of the recording, e.g., vinyl, 33 rpm, 8-track tape, cassette, compact disc, mp3, ogg vorbis. The howpublished field, newly included for the 17th edition, can hold similar information "for streaming audio formats and downloads" (14.263, 15.57). It can also, alternatively, hold the name of the streaming service, e.g., Spotify (cf. rihanna:umbrella).

The entries in notes-test.bib should at least give you a good idea of how this all works, and that file also contains an example of an audio book presented in a music entry. If you browse the examples in the *Manual* you will see the sheer variety of possibilities for presenting these sources, my intention being that judicious manipulation of .bib entries should allow you to make biblatex-chicago do what you want. Please let me know if I've ignored something you need. (Cf. 14.263–64, 15.57; eventdate, origdate, userd; \DeclareLabeldate in section 5.4.1 and avdate in section 5.4.2; auden:reading, beethoven:sonata29, bernstein:shostakovich, floyd:atom, holiday:fool, nytrumpet:art, rubinstein:chopin.)

mvbook
mvcollection
mvproceedings
mvreference All four of these entry types function more or less as in standard biblatex. I would like, however, to emphasize a couple of things. First, each is aliased to the entry type that results from removing the "**mv**" from their names. Second, each has an important role as the target of cross-references from other entries, the title of the **mv*** entry *always* providing a maintitle for the entry referencing it. If you want to provide a booktitle for the referencing entry, please use another entry type, e.g., **collection** for **incollection** or **book** for **inbook**. (These distinctions are particularly important to the correct functioning of the abbreviated references that biblatex-chicago, in various circumstances, provides. Please see the documentation of the **crossref** field in section 5.2, below.)

On the same subject, when multi-volume works are presented in the reference apparatus, the *Manual* (14.116–22, 15.41) requires that any dates presented should be appropriate to the specific nature of the citation. In short, this means that a date range that is right for the presentation of a multi-volume work in its entirety isn't right for citing, e.g., a single

volume of that work which appeared in one of the years contained in the date range. Because child entries will by default inherit all the date fields from their parent (including the endyear of a date range), I have turned off the inheritance of date and origdate fields from all of the **mv*** entry types to any other entry type. When the dates of the parent and of the child in such a situation are exactly the same, then this unfortunately requires an extra field in the child's .bib entry. When they're not the same, as will, I believe, often be the case, this arrangement saves a lot of annoying work in the child entry to suppress wrongly-inherited fields. Other sorts of parent entries aren't affected by this. See harley:ancient:cart, harley:cartography, and harley:hoc for how this might look.

Finally, in order to cope with parts of the notes & bibliography specification I have fairly thoroughly revised the **mv*** types there for the 17th edition. The author-date specification is, as I read it, simpler, so I haven't revised these types here. If you should happen to leave a stray `maintitle relatedtype` in an entry when switching specifications that entry will use the `multivolume relatedtype` instead, which will very likely not be what you want.

online   One of the features of the 17th edition of the *Manual* is the considerably extended, but still scattered, treatment of online materials (8.189–92, 14.6–18, 14.159–63, 14.175–76, 14.187, 14.189, 14.205–10, 14.233, 15.4, 15.9, 15.49–52, 15.57). The principles of that treatment have changed somewhat, as the *Manual* now places greater emphasis on the *location* of a source, which can in many cases outweigh, as far as choosing an entry type goes, the *nature* of the source. Working out the correspondences between online sources and biblatex-chicago entry types can, therefore, be tricky, so I have included table 2 summarizing the increasingly detailed instructions in the *Manual*, along with some further annotations here that might help to clarify it.

The basic principle, as I've cited in the penultimate entry of table 2, is that "the title of a website that is analogous to a traditionally printed work but does not have (and never had) a printed counterpart can be treated like the titles of other websites, subject to editorial discretion" (14.206). This means that an intrinsically online entry like stenger:privacy (citing CNN.com) need no longer be an article but can be presented in an online entry. (The same principle applies to wikiped:bibtex, but because of the code facilitating presentation of alphabetized entries in reference works, it's best in this case to keep the inreference entry but add an entrysubtype so that the title is presented as it would be in an online entry.) The corollary of the principle, as the first entry in table 2 suggests, is that an online edition of a printed work will generally require the same entry type as that printed work itself would. Blogs are, therefore, somewhat anomalous in requiring the various periodical types, though the *Manual* does specify that if you're not sure whether a website is a blog, then it probably requires the online type (14.206). Social media, on the other hand, are very much subject to the first principle, requiring online entries no matter whether the citation is of text, a photo, or a video. Without pretending that all of the correspondences flow deductively from the basic principles, I hope that the table might simplify most of your choices. If something remains unclear, please let me know and I'll see if I can improve it.

A few more notes are in order. I designed the new `relatedtype commenton` to facilitate citation of online comments, and it is available in two entry types, online and review. In both types the *Manual* (15.51–52) recommends that such material appear *only* in the text and not in the reference list, but I have attempted to simplify the presentation of such material wherever you want it to appear. Following the specifications, then, the default when you use `commenton` is for biblatex-chicago-authordate to modify how your .bib entry appears in the .bbl file by setting both `skipbib` and `cmsdate=full` in the options field, so that nothing appears in the reference list and citations present the full date and possibly also a time stamp (see below). Further, the style sets the verbc field so that these entries don't interfere with the provision of extra date letters — the full date and time should be enough to individuate separate comments. Finally, the style creates a new customc entry in your .bbl file which you can cite after your initial `commenton` entry using `\autocites` and which will, as a comment to your initial entry, say whether it's a comment or a reply or what have you, and then giving the short citation of that upon which it is a comment.

Table 2: Online materials and author-date entry types

Online Material	Entry Type	CMS Ref.	Sample Entry	Notes
Online edition of trad. publ. matter.				Use the same entry type as you would choose were you citing it from a printed source.
	@Book	14.161-62	james:ambassadors	CMS prefers (scanned) original page nos.
	@Article	14.175	black:infectious	If no "suitable URL" is available, e.g., if it
	@Review	15.47-49		points to a generic portal page rather than to an abstract, use the name of the commercial database in an addendum field instead.
Blogs		15.51		
Single post	@Article		amlen:hoot	The maintitle field holds the larger publication of which the blog is a part.
Whole Blog	@Periodical		amlen:wordplay	This & the next usually not in the ref. list.
Comment	@Review		viv:amlen	The commenton relatedtype helps manage these, in or out of the reference list.
Social Media	@Online	15.52		This includes anything — posts, photos, videos — on these and similar sites; the *location* of the material defines its treatment.
Mailing list or forum post		14.210	powell:email	Posts on private lists are to be treated as "personal communications," using @Misc w/ entrysubtype.
Facebook		14.209	diaz:surprise	
Twitter			obrien:recycle	
Instagram			souza:obama	
Comments / replies		14.210	braun:reply	This and the next are usually not included in the reference list. The commenton relatedtype helps manage them, in or out of that list.
		14.209	licis:diazcomment	
Online Multimedia		15.57		
Online video	@Online	14.267	pollan:plant	This category includes TED talks and most informal videos on YouTube and similar sites.
Online video, from a trad. journal	@Article		kessler:nyt	You can use @Online, but this requires special formatting in the note or titleaddon field.
Published films in an archive	@Video		weed:flatiron	
Podcasts	@Audio		danforth:podcast	Note the eventdate of the individual episode.
Archival audio	@Misc w/ entrysubtype	14.264	roosevelt:speech	Can have both a venue and an origlocation.
Streaming Media				
TV / Film	@Video	14.265	mayberry:brady	The streaming service is supplied by the URL. The tvepisode entrysubtype is new in the 17th edition.
Music	@Music	14.263	rihanna:umbrella	The streaming service is supplied by the howpublished field.
News / Interviews	@Article @Review	14.213	bundy:macneil	Network information goes in the usera field.
Websites	@Online	14.206-7 15.50	evanston:library stenger:privacy	An online source "analogous to a traditionally printed work but [which] does not have (and never had) a printed counterpart" may now use an @Online entry, at your discretion. If you only have an access date, "n.d." will appear as the publication date.
Reference works, cited by alphabetized entry	@InReference w/ entrysubtype	14.233	wikiped:bibtex	As above, you can choose the @Online treatment of the title, but it's best achieved using an @InReference entry w/ entrysubtype.
Scientific databases	@Dataset	14.257	genbank:db	Treated as author-title by default.

As an example, take the Facebook post diaz:surprise, which does appear in the reference list. The entry licis:diazcomment presents a comment on this post using the relatedtype commenton, so biblatex-chicago-authordate creates a new entry, diaz:surprise-customc, which won't appear in your .bib file (which is never altered) but in the .bbl file that biber produces to supply biblatex with the data for its citations. A command like `\autocites{licis:diazcomment}{diaz:surprise-customc}` will produce a citation like (Licis, February 24, 2016; comment on Díaz 2016). You can alter the string connecting the two citations (by default `\bibstring{commenton}`) by using the relatedstring field in the first of them (cf. powell:comment). (Note how minimal the .bib entry of a comment using this system can be — author, related, relatedtype, and date are pretty much the only fields required.)

Those who want online comments to appear in the reference list can still use the commenton relatedtype, and the same citation of the commented piece will appear there, connected by the same string that the customc entry provides. Here, though, you can also provide a separate title for the comment, and/or a separate url for it, should they exist, which will be printed before/after the citation of the commented piece, respectively. (In review entries, which use the same relatedtype, only the generic title is available, as is always the case with such entries.) If you manually set either (or both) of the cmsdate or the skipbib options in your entry then biblatex-chicago will assume you want to hand-craft that entry without its intervention, though it will still provide the virtual customc entry in your .bbl file, as that may still prove convenient. Note also that any verbc field you provide will never be altered by the package.

In general, constructing an online .bib file entry is much the same as in biblatex. The title field would contain the title of the page, the organization field could hold the title or owner of the whole site. If there is no specific title for a page, but only a generic one, then such a title should go in titleaddon, not forgetting to begin that field with a lowercase letter so that capitalization will work out correctly. It is worth remarking here, too, that the *Manual* (15.50) strongly prefers, if they're available, revision dates to access dates when documenting online material. If there is only a urldate in an entry, and that date is an access date (i.e., there's no userd field), then "n.d." will appear in citations and at the head of the entry in the reference list. Moreover, given how rapidly online sources can change (14.191, 14.209, 14.233), a time stamp may often be necessary further to specify a revision date (urldate) or the date of a comment or reply (date). This time specification should be added to the date field using biblatex's standard format, i.e., 2008-07-01T10:18:00. If a time zone is needed, then a separate timezone or urltimezone field is the best way, as it allows you to provide the initialisms that the *Manual* prefers (10.41, 14.191). On all of this please see date, timezone, urldate, userd, and verbc in section 5.2, below. Cf. also the documentation of the commenton relatedtype in section 5.2.1.

patent The *Manual* is very brief on the subject of patents (15.55), but very clear about which information it wants you to present, so such entries may not work well with other biblatex styles. Chicago's author-date style prefers the *later* of the two possible dates to appear in citations and at the head of the entry in the list of references. If a patent has been filed but not yet granted, then you can place the filing date in either the date field or the origdate field, and biblatex-chicago-authordate will automatically prepend the bibstring patentfiled to it. If the patent has been granted, then you put the filing date in the origdate field, and you put the date it was issued in the date field, to which the bibstring patentissued will automatically be prepended, and it is this later date that will head the entry and appear in citations. The patent number goes in the number field, and you should use the standard biblatex bibstrings in the type field. Though it isn't mentioned by the *Manual*, biblatex-chicago-authordate will print the holder after the author, if you provide one. Finally, the style capitalizes the title sentence-style, which seems to be the generally-accepted convention across all Chicago specifications. If you need to keep a word capitalized then you should wrap it in curly braces. See petroff:impurity.

performance The 17th edition of the *Manual* includes a new section (14.266) on citing live performances, and even though such references can usually be limited to the main text (cf. 15.57) it may sometimes be useful to include them in a reference list. Since biblatex provides the **performance** type, albeit without using it in its standard styles, I though it might be useful to

define it for biblatex-chicago, particularly as the other option for such material is the misc entry without any entrysubtype, and that entry type is already somewhat overloaded, though you can still use it if you wish.

Such entries will generally have a title, a venue, a location for the venue, and a date for the performance, along with a possible plethora of authorial and/or editorial roles depending on which sorts of contributor(s) you wish to emphasize in the citation. The editor[abc] and editor[abc]type fields should be most helpful here. I have included strings for choreographer in all localization files, but for others you may need to provide them in the editor[abc]type fields as you wish them printed — biblatex-chicago will automatically capitalize any that start with a lowercase letter. For the author-date styles it will probably be convenient to allow one of these names to appear at the head of the entry and in citations, as this will facilitate the appearance of the extradate to distinguish, e.g., multiple performances of the same work or performances of different works by the same producer or choreographer.

periodical This is the standard biblatex entry type for presenting an entire issue of a periodical, rather than one article within it. It has the same function in biblatex-chicago, and in the main uses the same fields, though in keeping with the system established in the article entry type (which see) you'll need to provide entrysubtype magazine if the periodical you are citing is a "newspaper" or "magazine" instead of a "journal." Also, remember that the note field is the place for identifying strings like "special issue," with its initial lowercase letter to activate the automatic capitalization routines, though this isn't strictly necessary in the author-date styles. (See *Manual* 14.187; good:wholeissue.)

It is worth noting a few things. First, the titleaddon field is now available in these entries, but as the title here is analogous to the journaltitle in article or review entries, the new jtitleaddon option (section 5.4.2) governs the punctuation separating the titleaddon from the title. Second, the special biblatex field shortjournal allows you to present shortened journaltitles in article, review, and periodical entries, as well as facilitating the creation of lists of journal abbreviations in the manner of a shorthand list. Because the periodical type uses the title field instead of journaltitle, biblatex-chicago automatically copies any shorttitle field, if one is present, into shortjournal. Please see the documentation of **shortjournal** in section 5.2 for all the details on how this works. Finally, although the 17th edition recommends that references to whole blogs, as opposed to individual blog posts, need appear only in the text (15.51), using the periodical type for such material can help with this, in or out of the reference list. The new authortitle entry option will ensure the presence of the name of the blog (as opposed to the non-existent date) in citations, and you could also use a \citeurl command to give the URL in the text (or a note). Alternately, you could let the entry appear in the reference list and cite it in the usual way. In that list the *Manual* (14.208) recommends that you include the name of any larger (usually periodical) publication of which the blog is a part. The maintitle field (with mainsubtitle and maintitleaddon, if needed) is the place for it. Cf. amlen:wordplay.

proceedings This is the standard biblatex and BibTeX entry type, now also including the eventdate, eventtitle, eventtitleaddon, and venue fields for identifying the event that produced the proceedings. The package can provide automatically abbreviated references in the reference list when you use a crossref or an xref field. The functionality is not enabled by default, but you can enable it in the preamble or in the options field using the booklongxref option. Please see **crossref** in section 5.2 and booklongxref in section 5.4.2, below.

reference This entry type is aliased to collection by the standard biblatex styles, but I intend it to be used in cases where you need to cite a reference work but not an alphabetized article or articles in that work. This could be because it doesn't contain such articles, and yet you still want the entry in the list of references to start with the title. Indeed, the only differences between it and inreference are the lack of a lista field to present an alphabetized entry, and the fact that any postnote field will be printed verbatim, rather than formatted as an alphabetized entry. (Cf. inreference, above.)

report   This entry type is a biblatex generalization of the traditional BIBTEX type techreport. Instructions for such entries are rather thin on the ground in the *Manual* (8.186, 14.220), so I have followed the generic advice about formatting it like a book, and hope that the results conform to the specification. At least one user has indicated a need, now filled, for an unpublished entrysubtype, which prints the title inside quotation marks (or, in authordate-trad, in plain roman) instead of in italics, but affects nothing else. This detail aside, the type's main peculiarities are the institution field in place of a publisher, the type field for identifying the kind of report in question, and the isrn field containing the International Standard Technical Report Number of a technical report. As in standard biblatex, if you use a techreport entry, then the type field automatically defaults to \bibstring{techreport}. As with booklet and manual, you can also use a book entry, putting the report type in note and the institution in publisher. (See herwign:office.)

review   As its name suggests, the review entry type was designed for reviews published in periodicals, and if you've already read the article instructions above — if you haven't, I recommend doing so now — you'll know that review serves as well for citing other sorts of material with generic titles, like letters to the editor, obituaries, interviews, online comments and the like. The primary rule is that any piece that has only a generic title, like "review of …," "interview with …," or "obituary of …," calls for the review type. Any piece that also has a specific title, e.g., "'Lost in BIBTEX,' an interview with …," requires an article entry. (This assumes the text is found in a periodical of some sort. Were it found in a book, then the incollection type would serve your needs, and you could use title and titleaddon there. While we're on the topic of exceptions, the *Manual* includes an example — 14.213 — where the "Interview" part of the title is considered a subtitle rather than a titleaddon, said part therefore being included inside the quotation marks and capitalized accordingly. Not having the journal in front of me I'm not sure what prompted that decision, but biblatex-chicago would obviously have no difficulty coping with such a situation.)

Once you've decided to use review, then you need to determine which sort of periodical you are citing, the rules for which are the same as for an article entry. If it is a "magazine" or a "newspaper", then you need an entrysubtype magazine, or the synonymous entrysubtype newspaper. The generic title goes in title and the other fields work just as as they do in an article entry with the same entrysubtype, including the substitution of the journaltitle for the author if the latter is missing. (See 14.190–91, 14.195–96, 14.201–4, 14.213, 15.49; barcott:review, bundy:macneil, Clemens:letter, gourmet:052006, kozinn:review, nyt:trevorobit, unsigned:ranke, wallraff:word.) If, on the other hand, the piece comes from a "journal," then you don't need an entrysubtype. The generic title goes in title, and the remaining fields work just as they do in a plain article entry. (See 14.202; ratliff:review.)

The *Manual* now suggests that, no matter which citation style you are using, it is "usually sufficient to cite newspaper and magazine articles entirely within the text" (15.47). This involves giving the title of the journal and the full date of publication in a parenthetical reference, including any other information in the main text (14.206), thereby obviating the need to present such an entry in the list of references. To utilize this method in the author-date styles, in addition to a magazine entrysubtype, you'll need to place cmsdate=full into the options field, including skipbib there as well to stop the entry printing in the list of references. If the entry only contains a date and journaltitle that's enough, but if it's a fuller entry also containing an author then you'll also need useauthor=false in the options field. Other surplus fields will be ignored. (See osborne:poison.)

Biblatex-chicago also, at the behest of Bertold Schweitzer, supports the relatedtype reviewof, which allows you to use the related mechanism to provide information about the work being reviewed, thereby simplifying how much information you need to provide in the reviewing entry. In particular, it relieves you of the need to construct title or titleaddon fields like: review of \mkbibemph{Book Title} by Author, as the related entry's title automatically provides the title in the review type and the titleaddon in the article type, with the related mechanism providing the connecting string. This may be particularly helpful if you need to cite multiple reviews of the same work. Please see section 5.2.1 for further information.

Most of the onerous details are the same as I described them in the **article** section above, but I'll repeat some of them briefly here. If anything in the title needs formatting, you need to provide those instructions yourself, as the default is completely plain. Authorless reviews are treated just like similar articles — with an entrysubtype, the journaltitle replaces the author in citations and heads the entry in the list of references, without an entrysubtype the title does the same. In the former case, Biber handles the sorting for you, but in the latter you'll need a sortkey because journaltitle comes before title in the sorting scheme. (14.204; gourmet:052006, nyt:trevorobit, unsigned:ranke, and see \DeclareSortingTemplate in section 5.4.1, below.). As in misc entries with an entrysubtype, words like "interview," "review," and "letter" only need capitalization after a full stop, so you can start the title field with a lowercase letter and let the automatic field formatting with \autocap do its work, though this isn't strictly necessary with biblatex-chicago-authordate.

A few details of the review type are fairly new, and in particular have changed between the 16th and 17th editions of the *Manual.* As I mentioned above, blogs are best treated as articles with magazine entrysubtype, whereas comments on those blogs — or replies to those comments, etc. — need the review type with the same entrysubtype. The 17th edition recommends that blog comments appear only in the text, and not in the reference list (15.51), so just as with comments in social media threads, for which see the **online** type above, I have provided the commenton relatedtype to simplify the presentation of such material wherever you want it to appear. Following the specifications, then, the default when you use commenton is for biblatex-chicago-authordate to modify how your .bib entry appears in the .bbl file by setting both skipbib and cmsdate=full in the options field, so that nothing appears in the reference list and citations present the full date and possibly also a time stamp (see below). Further, the style sets the verbc field so that these entries don't interfere with the provision of extra date letters — the full date and time should be enough to individuate separate comments. Finally, the style creates a new customc entry in your .bbl file which you can cite after your initial commenton entry using \autocites and which will, as a comment to your initial entry, say whether it's a comment or a reply or what have you, and then giving the short citation of that upon which it is a comment.

As an example, take the blog ellis:blog, which does appear in the reference list. The entry ac:comment presents a comment on this post using the relatedtype commenton, so biblatex-chicago-authordate creates a new entry, ellis:blog-customc, which won't appear in your .bib file (which is never altered) but in the .bbl file that biber produces to supply biblatex with the data for its citations. A command in your document like \autocites{ac:comment}{ellis:blog-customc} will produce a citation like (AC, July 1, 2008, 10:18 a.m.; comment on Ellis 2008). You can alter the string connecting the two citations (by default \bibstring{commenton}) by using the relatedstring field in the first of them. (Note how minimal the .bib entry of a comment using this system can be — author, entrysubtype, related, relatedtype, and eventdate are pretty much the only fields required.)

Those who want online comments to appear in the reference list can still use the commenton relatedtype, and the same citation of the commented piece will appear there, connected by the same string that the customc entry provides. Here, though, you can also provide a separate url for the comment, should it exist, which will be printed after the citation of the commented piece. (In online entries, which use the same relatedtype, you can also provide a separate title for the comment.) If you manually set either (or both) of the cmsdate or the skipbib options in your entry then biblatex-chicago will assume you want to hand-craft that entry without its intervention, though it will still provide the virtual customc entry in your .bbl file, as that may still prove convenient. Note also that any verbc field you provide will never be altered by the package. (Please see the documentation of this relatedtype in section 5.2.1, that of verbc in section 5.2, and also the information about online materials in table 2.)

The new edition of the *Manual* retains the requirement for a date closely associated with the comment (14.208, 15.51), so in such entries you now have a choice. If you are using the commenton relatedtype, you can use the date or eventdate indifferently, as even

when you print the entry in the reference list the reference to the main blog provides its own date. If, in 16th-edition style, you print a fuller entry in the reference list, then you'll need the eventdate for the comment, as the date applies to the main blog post. If you need a time stamp in addition, as may frequently be the case with multiple contributions by the same author to a single thread, then you should now use the standard biblatex time-stamp format (e.g., 2008-07-01T10:18:00) in whichever of the two date fields you're using, and not, as previously, in the nameaddon field. Biblatex-chicago will format and print it appropriately. This change allows the nameaddon field to revert to its primary use, which is to provide extra information about the author. In blog comments, this could include the commenter's geographical location, which you need to enclose in parentheses, as I've removed the automatic square brackets from this field to allow it this more general usefulness. You can, of course, still provide your own square brackets in review entries to indicate pseudonymous authorship, which is the standard function of nameaddon in most entry types. The package options nameaddonformat and nameaddonsep can help here, as well. See sections 5.4.3 and 5.4.2, below. Please also see the documentation of date, eventdate, and timezone in section 5.2, \DeclareLabeldate in section 5.4.1, and avdate in section 5.4.2.

For the reasons I explained in the article docs above, I have brought the article and review entry types into line with most of the other types in allowing the use of the namea and nameb fields in order to associate an editor or a translator specifically with the title. The editor and translator fields, in strict homology with other entry types, are associated with the issuetitle if one is present, and with the title otherwise. The usual string concatenation rules still apply — cf. editor and editortype in section 5.2, below.

Finally, the special biblatex field shortjournal allows you to present shortened journaltitles in review entries, as well as in article and periodical entries, and it facilitates the creation of lists of journal abbreviations in the manner of a shorthand list. Please see the documentation of **shortjournal** in section 5.2 for all the details on how this works.

standard   In older releases it was fairly straightforward to present published national or international standards using a book entry, but with some additional specifications now included in the 17th edition of the *Manual* (14.259, 15.37) I think it might be helpful to provide a separate entry type. The **standard** type has long existed in biblatex, though none of its included styles use it. In biblatex-chicago constructing such an entry is mostly straightforward. The organization responsible for the standard goes in organization, the title in title, and the series and number fields provide the ID of the standard. The date field generally provides the publication date, though for some standards there may also be a later reaffirmation date (or similar), for which you can use the eventdate. To choose which year appears in citations, the **standard** type follows, by default, the same ordering as review and music entries, so that the eventdate will, if present, provide the year. (Cf. avdate in section 5.4.2, below.)

Now, for the peculiarities. In the reference list, the organization will appear at the head of the entry, and will be reprinted as the publisher. If you wish to provide a shortened version for the second appearance, then the publisher field is the place for it. You can also use an author instead of an organization, but in such a case you'll have to provide a publisher, and no matter which field you choose to appear at the head of the entry you'll usually have to think about providing some sort of abbreviated form for citations. A shortauthor will appear only in citations, while a shorthand can also appear at the head of the entry in the list of references. (If you provide the latter, biblatex-chicago will automatically sort entries by it.) Any named editor or namec will, as per the specification, *not* appear at the head of entries. You can really only alter this by using a book entry, instead. (Cf. w3c:xml, and the shorthand docs on page 138, below.)

Finally, it is distinctly possible that an entry with two dates will need somehow to specify just what sort of dates are involved. The usual biblatex-chicago method is the userd field, and here that field will act as a date-type for the date field itself, assuming as usual that there is no urldate. For the eventdate, you'll need to use howpublished, which I have commandeered for this purpose in a few other entry types, as well. (Cf. niso:bibref and **howpublished** in section 5.2, below.)

suppbook This is the entry type to use if the main focus of a reference is supplemental material in a book or in a collection, e.g., an introduction, afterword, or forward, either by the same or by a different author. There are two mechanisms in biblatex-chicago for producing such a citation. First, these three just-mentioned types of material, and only these three types, can be referenced using the introduction, afterword, or foreword fields, a system that requires you simply to define one of them in any way and leave the others undefined. The macros don't use the text provided by such an entry, they merely check to see if one of them is defined, in order to decide which sort of pre- or post-matter is at stake, and to print the appropriate string before the title in the list of references, and possibly also in the list of shorthands. This mechanism works without modification across multiple languages, but I have also provided functionality which allows you to cite any sort of supplemental material whatever, using the type field. Under this second system, simply put the nature of the material, including the relevant preposition, in that field, beginning with a lowercase letter so biblatex can decide whether it needs capitalization depending on the context. Examples might be "preface to" or "colophon of." (Please note, however, that unless you use a \bibstring command in the type field, the resultant entry will not be portable across languages.)

The other rules for constructing your .bib entry remain the same. The author field refers to the author of the introduction or afterword, while bookauthor refers to the author of the main text of the work, if the two differ. Recent editions of the *Manual* requires that you include the page range for the cited part in the list of references. As ever, if the focus of the reference is the main text of the book, but you want to mention the name of the writer of an introduction or afterword for completeness, then the normal biblatex rules apply, and you can just put their name in the appropriate field of a book entry, that is, in the foreword, afterword, or introduction field. (See *Manual* 14.110; friedman:intro, polakow:afterw, prose:intro).

suppcollection This fulfills a function analogous to suppbook. Indeed, I believe the **suppbook** type can serve to present supplemental material in both types of work, so this entry type is an alias to suppbook, which see.

suppperiodical This type is intended to allow reference to generically-titled works in periodicals, such as regular columns or letters to the editor. Biblatex also provides the review type for this purpose, so in both Chicago styles suppperiodical is an alias of review. Please see above under **review** for the full instructions on how to construct a .bib entry for such a reference.

unpublished The unpublished entry type works largely as it does in standard biblatex, though it's worth remembering that you should use a lowercase letter at the start of your note field (or perhaps an \autocap command in the somewhat contradictory howpublished, if you have one) for material that wouldn't ordinarily be capitalized except at the beginning of a sentence. Thanks to a bug report by Henry D. Hollithron, such entries will print information about any editor, translator, compiler, etc., that you include in the .bib file. Also, conforming to the indications of the *Manual*, and thanks to the prompting of Jan David Hauck, you can use the venue, eventdate, eventtitle, and eventtitleaddon fields further to specify unpublished conference papers and the like (14.216–18; nass:address).

video This is the last of the three audiovisual entry types, and as its name suggests it is intended for citing visual media, be it films of any sort or TV shows, broadcast, on the Net, on VHS, DVD, or Blu-ray, though it will serve as well, I think, for radio broadcasts of plays or drama serials. As with the music type discussed above, certain choices had to be made when associating the production roles found, e.g., on a DVD, to those bookish ones provided by biblatex. The 17th edition of the *Manual* once again revised its recommendations for this type, but fortunately the changes are additive, i.e., you can re-use 16th-edition citations but are encouraged to peruse the following guidelines to see if there's any information you might think of adding to bring your citations more into line with the spec. Here are the main guidelines:

> author: This will not infrequently be left undefined, as the director of a film should be identified as such and therefore placed in the editor field with the appropriate

editortype (see below). You will need it, however, to identify the composer of, e.g., an oratorio on VHS (handel:messiah), or perhaps the provider of commentaries or other extras on a film DVD (cleese:holygrail).

editor, editora, editorb: The director or producer, or possibly the performer or conductor in recorded musical performances. These will ordinarily follow the title of the work, though the usual useauthor and useeditor options can alter the presentation within an entry. Because these are non-standard roles, you will need to identify them using the following:

editortype, editoratype, editorbtype: The most common roles, all associated with specific bibstrings (or their absence), will likely be director, producer, and, oddly, none. The last is particularly useful if you want to identify performers, as they usually don't need further specifying and this role prevents biblatex from falling back on the default editor bibstring. Any other roles you want to emphasize, even if there is no pre-defined \bibstring, can be provided here, and will be printed as-is, contextually capitalized. (Cf. hitchcock:nbynw.)

title, titleaddon, booktitle, booktitleaddon, maintitle: As with the other 2 audiovisual types, video serves as an analogue both to books and to collections, so the title may be of a whole film DVD or of a TV series, or it may identify one episode in a series or one scene in a film. In the latter cases, the title of the whole would go in booktitle. The booktitleaddon field is the place for specifying the season and/or episode number of a TV series, while the titleaddon is for any information that needs to come between the title and the booktitle (american:crime, cleese:holygrail, friends:leia, handel:messiah, hitchcock:nbynw, mayberry:brady). As in the music type, maintitle may be necessary for a boxed set or something similar.

entrysubtype: If, for some reason, you want to cite an individual episode or scene without reference to any larger unit, then your entry will contain only a title, which biblatex-chicago would normally interpret as the title of a complete film or TV series. In such a case, you'll need to define an entrysubtype to let it know that the lone title is such a sub-unit. In quite a different syntactic transformation, the 17th edition (14.265) now recommends that, when presenting episodes from a TV series, the name of the series (booktitle) comes before the episode name (title). The exact string tvepisode in the entrysubtype field achieves this reversal, which includes using the booktitle as a sorttitle in the reference list and also as the labeltitle in short notes.

date, eventdate, origdate, pubstate: The 17th edition of the *Manual* continues to encourage writers to find some way of dating audiovisual materials, while if there is more than one date "the date of the original recording should be privileged" (15.57). As with music entries, in order to follow these specifications I have had to provide three separate date fields for citing video sources, but their uses differ somewhat between the two types. In both, the date will generally provide the publishing or copyright date of the medium you are referencing. More specific to this entry type, the origdate will generally hold the date of the original theatrical release of a film, while the eventdate will most commonly present either the broadcast date of a particular TV program, or the recording/performance date of, for example, an opera on DVD. The style will automatically prepend the bibstring broadcast to such a date, though you can use the userd field to change the string printed there. (Absent an eventdate, the userd field in video entries will modify the urldate, and absent those two it will modify the date.) Typically, any given video entry will only need an eventdate *or* an origdate, and it is this date that will appear in citations and at the head of the entry in the reference list. It's conceivable that you may need all three dates, in which case you can also use the standard pubstate field with reprint in it to control the printing of the origdate at the end of the entry, though I have altered the string that is printed there. Cf. friends:leia, handel:messiah, hitchcock:nbynw; pubstate, below.

type: As in all the audiovisual entry types, the type field holds the medium of the title, e.g., 8 mm, VHS, DVD, Blu-ray, MPEG.

As with the music type, entries in dates-test.bib should at least give you a good idea of how all this works. (Cf. 14.265, 14.267; eventdate, origdate, userd; \DeclareLabeldate in section 5.4.1, and avdate in section 5.4.2; cleese:holygrail, friends:leia, handel:messiah, hitchcock:nbynw, loc:city, weed:flatiron.)

5.2 Entry Fields

The following discussion presents, in alphabetical order, a complete list of the entry fields you will need to use biblatex-chicago-authordate. As in section 5.1, I shall include references to the numbered paragraphs of the *Chicago Manual of Style*, and also to the entries in dates-test.bib. Many fields are most easily understood with reference to other, related fields. In such cases, cross references should allow you to find the information you need.

addendum   As in standard biblatex, this field allows you to add miscellaneous information to the end of an entry, after publication data but before any url or doi field. In the patent entry type (which see), it will be printed in close association with the filing and issue dates. In any entry type, if your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and the style will take care of the rest. Cf. note. (See *Manual* 14.114, 14.159–63; davenport:attention, natrecoff:camera.)

afterword   In most circumstances, this field will function as it does in standard biblatex, i.e., you should include here the author(s) of an afterword to a given work. The *Manual* suggests that, as a general rule, the afterword would need to be of significant importance in its own right to require mentioning in the reference apparatus, but this is clearly a matter for the user's judgment. As in biblatex, if the name given here exactly matches that of an editor and/or a translator, then biblatex-chicago will concatenate these fields in the formatted references.

As noted above, however, this field has a special meaning in the suppbook entry type, used to make an afterword, foreword, or introduction the main focus of a citation. If it's an afterword at issue, simply define afterword any way you please, leave foreword and introduction undefined, and biblatex-chicago will do the rest. Cf. foreword and introduction. (See *Manual* 14.105, 14.110; polakow:afterw.)

annotation   At the request of Emil Salim, biblatex-chicago provides a package option (see annotation below, section 5.4.3) to allow you to produce annotated reference lists. A recent feature request by Moritz Wemheuer referenced a StackExchange question which suggested that the possible uses for the annotation field could well be more extensive, appearing as it does at the very end of all entry types. I have therefore modified the annotation option so that you can print the field in the reference list (=bib or =true, the default), in long (probably legal) notes (=notes), in both (=all), or in neither (=false). The two options bibannotesep and citeannotesep allow you to choose the separator between the rest of the entry and the annotation, and to choose a different one in reference list and notes. The default formatting in the reference list (vpar) is to print the annotation as a separate block using \par\nobreak\vskip\bibitemsep #1, while in long notes the default (period) is to print it simply as an additional field, separated by a period. The *Manual's* guidelines (14.64) allow for both these possibilities, and I have provided a range of others, for which you should consult the full documentation in section 5.4.2. (Note also that both options can be set globally or per-type in the preamble, or per-entry in the options field of individual entries. For specialized needs, of course, you can re-declare the format [\DeclareFieldFormat{annotation}] in your preamble, or redefine the \bibannotesep and \citeannotesep commands there.) In section 5.4.3 you will find instructions for employing the new formatbib and entrybreak options to give you fine-grained control over the formatting of the entire reference list, particularly with regard to TEX's page-breaking algorithms. The aim is to remove, in most cases, any need for you to delve into the low-level commands involved in these algorithms.

annotator   I have implemented this biblatex field pretty much as that package's standard styles do, even though the *Manual* doesn't actually mention it. It may be useful for some purposes. Cf. commentator.

author For the most part, I have implemented this field in a completely standard fashion. Remember that corporate or organizational authors need to have an extra set of curly braces around them (e.g., `{{Associated Press}}`) to prevent biber from treating one part of the name as a surname (14.84, 14.200, 15.37; assocpress:gun, chicago:manual). If there is no author, then biblatex-chicago will look, in order, for a namea, an editor, a nameb, a translator, or a namec (i.e., a compiler) and use that name (or those names) instead, followed by the appropriate identifying string (esp. 15.36, also 14.76, 14.103, 14.121, 14.126, 14.180; boxer:china, brown:bremer, harley:cartography, schellinger:novel, sechzer:women, silver:gawain, soltes:georgia). Biber and biblatex take care of alphabetizing entries no matter which name appears at their head. In citations, where the labelname is used, the order searched is somewhat augmented: shortauthor, author, shorteditor, namea, editor, nameb, translator, and namec.

If you wish to emphasize the activity of an editor, a translator, or a compiler (14.104; eliot:pound), you can use the biblatex options useauthor=false, usenamea=false, useeditor=false, usenameb=false, usetranslator=false, and usenamec=false in the options field to choose which name appears at the head of an entry and in the citation. You only need to turn off any fields that are present in the entry, but please remember to use the new option usenamec instead of the old usecompiler (which I've deprecated), as the latter doesn't work as smoothly and completely as biblatex's own name toggles. See \DeclareSortingTemplate in section 5.4.1, and the editortype documentation, below.

Of course, in collection and proceedings entry types, an author isn't expected, so there the chain of substitutions begins with namea and editor. Also, in article and review entries with entrysubtype magazine, the absence of an author triggers the use of the journaltitle in its stead. Without an entrysubtype, the title will be used. See the discussion a few paragraphs down, and those entry types, for further details.

Recommendations concerning anonymous authors in other kinds of references (15.34) emphasize using the title in citations and at the head of reference list entries, rather than "Anonymous." The latter may still in some cases be useful "in a bibliography in which several anonymous works need to be grouped" (14.79), but even with a source like virginia:plantation, "the reference list entry should normally begin with the title… Text citations may refer to a short form of the title but must include the first word (other than an initial article)" (15.34). The shorttitle field is the place for the short form, and you'll also need a sortkey of some sort if the full title begins with an article that is to be ignored when alphabetizing.

If "the authorship is known or guessed at but was omitted on the title page," then you need to use the authortype field to let biblatex-chicago know this fact (15.34). If the author is known (horsley:prosodies), then put anon in the authortype field, if guessed at (cook:sotweed) put anon? there. (In both cases, biblatex-chicago tests for these *exact* strings, so check your typing if it doesn't work.) This will have the effect of enclosing the name in square brackets, with or without the question mark indicating doubt. As long as you have the right string in the authortype field, biblatex-chicago-authordate will also do the right thing automatically in text citations.

In most entry types (except customc) the nameaddon field furnishes the means to cope with the case of pseudonymous authorship. If the author's real name isn't known, simply put pseud. (or \bibstring{pseudonym}) in that field (centinel:letters). If you wish to give a pseudonymous author's real name, simply include it there, formatted as you wish it to appear, as the contents of this field won't be manipulated as a name by biblatex (lecarre:quest, stendhal:parma). If you have given the author's real name in the author field, then the pseudonym goes in nameaddon, in the form Firstname Lastname, pseud. (creasey:ashe:blast, creasey:morton:hide, creasey:york:death). This latter method will allow you to keep all references to one author's work under different pseudonyms grouped together in the list of references, a method recommended by the *Manual*. The *Manual* (14.82) recommends using cross-references from author to pseudonym or vice versa, so in these latter examples I have included such references from the various pseudonyms back to the author's name, using the customc entry type, which see (ashe:creasey, morton:creasey, york:creasey). Please see the entry on **nameaddon**, below, for circumstances where you may need to provide your own square brackets when presenting a pseudonym,

and also the package options `nameaddonformat` and `nameaddonsep` in sections 5.4.3 and 5.4.2, below.

As its name suggests, the author-date style very much wants to have a name of some sort present both for the entries in the list of references and for the in-text citations. The *Manual* is nothing if not flexible, however, so with unsigned articles or encyclopedia entries the journaltitle or title may take the place of the author (gourmet:052006, lakeforester:pushcarts, nyt:trevorobit, unsigned:ranke, wikipedia:bibtex). Even in such entries, however, it may be advantageous to provide either a standard shorttitle or, for abbreviating a journaltitle, a shortjournal field, thereby keeping the in-text citations to a reasonable length, though not at the expense of making it hard to find the relevant entries in the reference list. An institutional author's name can also be rather too long for in-text citations. In unsigned:ranke I placed an abbreviated form of the journaltitle into shortjournal, adapting for a periodical the practice recommended for books in 15.37. In iso:electrodoc, I provided a shorthand field, which by default in biblatex-chicago-authordate will appear both in text citations and at the head of the entry in the list of references, followed, within the entry, by its expansion, this latter placed within parentheses. Please see under **shorthand** below for the details. (You can utilize the list of shorthands to clarify the abbreviation, if you wish, and you can also provide a separate list of journal abbreviations using the `\printbiblist{shortjournal}` command. Please cf. the **shortjournal** documentation, below, and the journalabbrev option in section 5.4.2.)

authortype   In biblatex-chicago, this field serves a function very much in keeping with the spirit of standard biblatex, if not with its letter. Instead of allowing you to change the string used to identify an author, the field allows you to indicate when an author is anonymous, that is, when their name doesn't appear on the title page of the work you are citing. As I've just detailed under author, the *Manual* generally discourages the use of "Anonymous" (or "Anon." as an author, though in some cases it may well be your best option. If, however, the name of the author is known or guessed at, then you're supposed to enclose that name within square brackets, which is exactly what biblatex-chicago does for you when you put either anon (author known) or anon? (author guessed at) in the authortype field. (Putting the square brackets in yourself doesn't work right, hence this mechanism.) The macros test for these *exact* strings, so check your typing if you don't see the brackets. Assuming the strings are correct, biblatex-chicago will also automatically do the right thing in citations. (See the author docs just above. Also *Manual* 15.34; cook:sotweed, horsley:prosodies.)

The *Manual* doesn't clarify how to treat multiple works by the same author, in one or more of which their name doesn't appear on the title page. By default, biblatex-chicago will, after the first appearance in the reference list, replace identical authors with the 3-em dash, regardless of any authortype field that may be present. If you want to distinguish between works certainly written by and works merely ascribed to a given author, then you can use the dashed option in the options field of individual entries, and possibly also a sortname, to get the results you want.

bookauthor   For the most part, as in biblatex, a bookauthor is the author of a booktitle, so that, for example, if one chapter in a book has different authorship from the book as a whole, you can include that fact in a reference (will:cohere). Keep in mind, however, that the entry type for introductions, forewords and afterwords (suppbook) uses bookauthor as the author of title (polakow:afterw, prose:intro).

bookpagination   This, a standard biblatex field, allows you automatically to prefix the appropriate string to information you provide in a pages field. If you leave it blank, the default is to print no identifying string (the equivalent of setting it to none), as this is the practice the *Manual* recommends for nearly all page numbers. Even if the numbers you cite aren't pages, but it is otherwise clear from the context what they represent, you can still leave this blank. If, however, you specifically need to identify what sort of unit the pages field represents, then you can either hand-format that field yourself, or use one of the provided bibstrings in the bookpagination field. These bibstrings currently are column, line, paragraph, page, section, and verse, all of which are used by biblatex's standard styles.

There are two points that may need explaining here. First, all the bibstrings I have just listed follow the Chicago specification, which may be confusing if they don't produce the strings you expect. Second, remember that bookpagination applies only to the pages field — if you need to format a citation's postnote field, then you must use pagination, which see (10.42–43, 14.147–56).

booksubtitle The subtitle for a booktitle. See the next entry for further information.

booktitle In the bookinbook, inbook, incollection, inproceedings, and letter entry types, the booktitle field holds the title of the larger volume in which the title itself is contained as one part. It is important not to confuse this with the maintitle, which holds the more general title of multiple volumes, e.g., *Collected Works*. It is perfectly possible for one .bib file entry to contain all three sorts of title (euripides:orestes, plato:republic:gr). You may also find a booktitle in other sorts of entries (e.g., book or collection), but there it will almost invariably be providing information for the traditional BIBTEX cross-referencing apparatus (prairie:state), which I discuss below (**crossref**). Such provision is now unnecessary, assuming you are using biber. The booktitle no longer takes sentence-style capitalization in authordate, though it does in authordate-trad.

booktitleaddon An annex to the booktitle. It will be printed in the main text font, without quotation marks. If your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago will automatically do the right thing. The package and entry options ptitleaddon and ctitleaddon (section 5.4.2) allow you to customize the punctuation that appears before the booktitleaddon field.

chapter This field holds the chapter number, mainly useful only in an inbook or an incollection entry where you wish to cite a specific chapter of a book (ashbrook:brain). It now also holds the track number of individual pieces of music, whether on a traditional format or on a streaming service (holiday:fool, rihanna:umbrella).

commentator I have implemented this biblatex field pretty much as that package's standard styles do, even though the *Manual* doesn't actually mention it. It may be useful for some purposes. Cf. annotator.

crossref Biblatex uses the standard BIBTEX cross-referencing mechanism, and has also introduced a modified one of its own (xref). The latter works as it always has, attempting to remedy some of the deficiencies of the traditional mechanism by ensuring that child entries will inherit no data at all from their parents. Section 2.4.1 of biblatex.pdf contains useful notes on managing cross-referenced entries, and section 3.15 explains some of the limitations of the traditional backends, which offer only a small subset of Biber's features. For the crossref field, when Biber is the backend, biblatex defines a series of inheritance rules which make it much more convenient to use. Appendix B of biblatex.pdf explains the defaults, to which biblatex-chicago has added several that I should mention here: incollection entries can inherit from book and mvbook just as they do from mvcollection entries; letter entries inherit from book, collection, mvbook, and mvcollection entries the same way an inbook or an incollection entry would; the namea, nameb, sortname, sorttitle, and sortyear fields, all highly single-entry specific, are no longer inheritable; and date and origdate fields are not inheritable from any of the **mv*** entry types.

Aside from these inheritance questions, the other main function of the crossref and xref fields in biblatex-chicago is as a trigger for the provision of abbreviated entries in the list of references. The *Manual* (15.42) specifies that if you cite several contributions to the same collection, all (including the collection itself) may be listed separately in the list, which the package does automatically, using the default inclusion threshold of 2 in the case both of crossref'ed and xref'ed entries. (The familiar \nocite command may also help in some circumstances.) In the reference list an abbreviated form will be appropriate for all the child entries. The biblatex-chicago-authordate package has always implemented these instructions, but only if you use a crossref or an xref field, and only

in incollection, inproceedings, or letter entries (on the last named, see just below). Recent releases have considerably extended this functionality.

First, I added five entry types — **book**, **bookinbook**, **collection**, **inbook**, and **proceedings** — to the list of those which use shortened cross references, and I provided two options — longcrossref and booklongxref, on which more below — which you can use in the preamble or in the options field of an entry to enable or disable the automatic provision of abbreviated references. (The crossref or xref field are still necessary for this provision, but they are no longer sufficient on their own.) The inbook type works exactly like incollection or inproceedings; in previous releases, you could use inbook instead of incollection to avoid the automatic abbreviation, the two types being otherwise identical. Now that you can use an option to turn off abbreviated references even in the presence of a crossref or xref field, I have thought it sensible to include this entry type alongside the others. (Cf. ellet:galena, keating:dearborn, lippincott:chicago, and prairie:state to see this mechanism in action in the reference list.)

The inclusion of **book**, **bookinbook**, **collection**, and **proceedings** entries fulfills a request made by Kenneth L. Pearce, and allows you to obtain shortened references to, for example, separate volumes within a multi-volume work, or to different book-length works collected inside a single volume. Such references are not part of the *Manual's* specification, but they are a logical extension of it, so the system of options for turning on this functionality behaves differently for these four entry types than for the other 4 (see below). In dates-test.bib you can get a feel for how this works by looking at bernhard:boris, bernhard:ritter, bernhard:themacher, harley:ancient:cart, harley:cartography, and harley:hoc.

A published collection of letters requires a somewhat different treatment (15.40). In the author-date style, the *Manual* discourages individual letters from appearing in the list of references at all, preferring that the "dates of individual correspondence should be woven into the text." If you have special reason to do so, however, you can still present individual published letters there (using the letter entry type), and they too can use the system of shortened references just outlined, even though the *Manual* doesn't explicitly require it. As with book, bookinbook, collection, inbook, incollection, inproceedings, and proceedings entries, the use of a crossref or xref field will activate this mechanism, assuming the preamble and entry options are set to enable it. (See white:ross:memo, white:russ, and white:total, for examples of the xref field in action in this way, and please note that the second of these entries is entirely fictitious, provided merely for the sake of example.)

longcrossref These options function, by default, asymmetrically. The first, longcrossref, generally controls the settings for the entry types more-or-less authorized by the *Manual*: inbook, incollection, inproceedings, and letter.

> **false:** This is the default. If you use crossref or xref fields in the four mentioned entry types, you'll get the abbreviated entries in the reference list.

> **true:** You'll get no abbreviated citations of these entry types in the reference list.

> **none:** This switch is special, allowing you with one setting to provide abbreviated citations not just of the four entry types mentioned but also of book, bookinbook, collection, and proceedings entries.

> **notes, bib:** These two options are carried over from the notes & bibliography style; here they are synonymous with false and true, respectively.

booklongxref The second option, booklongxref, controls the settings for book, bookinbook, collection, and proceedings entries:

> **true:** This is the default. If you use crossref or xref fields in these entry types, by default you will *not* get any abbreviated citations in the reference list.

> **false:** You'll get abbreviated citations in these entry types in the reference list.

> **notes, bib:** These two options are carried over from the notes & bibliography style; here they are synonymous with false and true, respectively.

Please note that you can set both of these options either in the preamble or in the options field of individual entries, allowing you to change the settings on an entry-by-entry basis.

Please further note that in previous releases of biblatex-chicago I recommended against using shorthand, reprinttitle and/or userf fields in combination with this abbreviated cross-referencing mechanism. I received, however a request from Alexandre Roberts to allow the shorthand to appear in the place of the abbreviated cross-reference as an additional space-saving measure, and one from Kenneth Pearce to permit the combination of the other two fields with crossref, as well. All three of these fields, in any combination, should just work in such circumstances in biblatex-chicago-authordate, though if you are using a list of shorthands then you may need to include skipbiblist in the options field of some entries to avoid duplicates. If you come across any problems or inaccuracies, please report them.

Finally, there is also an xrefurl option available to control the printing of url, doi, and eprint fields in abbreviated references where such information might otherwise never appear. See xrefurl in section 5.4.3.

date   Predictably, this is one of the key fields for the author-date styles, and one which, as a general rule, every .bib entry designed for this system ought to contain. So important is it, that biblatex-chicago-authordate will, in most entry types, supply a missing \bibstring{nodate} if there is no date otherwise provided (15.44), or if there is only a urldate, and that date is an access date, i.e., there's no userd field (15.50). Citations will look like (Author, n.d.), and entries in the list of references will begin: Author, Firstname. n.d. This seems simple enough, but there are a surprising number of complications which require attention.

To start, in each entry, Biber attempts to find something which it can designate a labeldate, which will, in general and ideally, be the year printed both in citations and at the head of the entry in the list of references. The search for the labeldate is governed by instances of the declaration \DeclareLabeldate, which cannot be set on an entry-by-entry basis, but rather only in a document preamble (or in files used by biblatex or its styles, like biblatex-chicago). The declaration can set a different search order according to entry type, but other differentiations are not currently possible. In all cases, guided by the instructions given by the \DeclareLabeldate instances, Biber will search each entry in the declared order, and the first match will provide the labeldate. Only when it finds no match at all will it fall back on \bibstring{nodate}. (In the misc and dataset types this automatic provision is turned off, as such material may not be expected in many standard cases to have a usable date provided.) You can prevent the appearance of \bibstring{nodate} throughout your document in all entry types with the option nodates=false when loading biblatex-chicago in your preamble, or you can set it in the options field of individual entries. (See section 5.4.2, below.)

The thing to keep in mind is that *only* for a labelyear will biblatex provide what it calls the extradate field, which means the alphabetical suffix (1978**a**) to differentiate entries with the same author and year. A style can print any year it wants in a citation, but only the labelyear comes equipped with an extradate. (It is also, by the way, the field that the sorting algorithm will use for ordering the list of references.) So the challenge, in a style wherein entries can contain more than one date, is to allow different dates to appear in citations and at the head of reference list entries, but to ensure that, as often as is possible, that date *is* the labeldate. This sounds simple, but in practice it requires a series of options for date presentation, and multiple iterations of the \DeclareLabeldate command. There are *two* standard search orders set up by default: in music, review, standard, suppperiodical, and video entries, the default order is eventdate, origdate, date, urldate, while in all other entry types the default is date, eventdate, origdate, urldate. I believe that these defaults work well for most reference lists, especially those that contain relatively few entries with multiple dates, but if they don't work for you then the following options can help.

In the case of music, review, standard, suppperiodical, and video entries, the avdate (i.e., audio-visual date) option, set to true by default, can be set to false in your preamble to return these entry types to the general defaults. Please see the documentation of the

entry types in section 5.1 above for the details of how multiple dates will be treated in such entries, and also see avdate in section 5.4.2, below. If you don't alter the avdate settings, the other settings I am about to describe won't apply to such entries. For the entry types not covered by the avdate option, the *Manual* (15.40) presents a fairly simple scheme for when a particular entry has more than one date, but I have been unable to make its implementation quite as straightforward. If a reprinted book, say, has both a date of publication for the reprint edition and an origdate for the original edition, then by default biblatex-chicago-authordate will use the date in citations and at the head of the entry in the reference list. If you inform biblatex-chicago that the book is a reprint by putting the string reprint in the pubstate field, then a notice will be printed at the end of the entry saying "First published 1898." With no pubstate field (and no cmsdate option), the algorithms will ignore the origdate.

cmsdate
in entry

If, for any reason, you wish the origdate to appear at the head of the entry, then your first option is to use the cmsdate toggle in the options field of the entry itself. This has 3 possible states relevant to this context, though there is a fourth state (full) which I shall discuss below:

1. cmsdate=both prints both the origdate and the date, using the *Manual's* standard format: (Author [1898] 1952) in parenthetical citations, Author (1898) 1952 outside parentheses, e.g., in the reference list.

2. cmsdate=off is the default, discussed above: (Author 1952).

3. cmsdate=on prints the origdate at the head of the entry in the list of references and in citations: (Author 1898). NB: The *Manual* no longer includes this among the approved options. If you want to present the origdate at the head of an entry, then generally speaking you should probably use cmsdate=both. I have nevertheless retained this option for certain cases where it has proved useful. The obsolete options new and old work like both.

In the first and third cases, if you put the string reprint in the pubstate field, then the publication data in the list of references will include a notice, formatted according to the specifications, that the modern edition is a reprint. In the third case, since the date hasn't yet been printed, this publication data will also include the date of the modern reprint.

Let us imagine, however, that your list of references contains another book by the same author, also a reprint edition: (Author [1896] 1974). How will these two works be ordered in the list of references? By the labelyear, in this case the year field, which appears first in the default definition (date, eventdate, origdate, urldate) of \DeclareLabeldate, and which in this case will be wrong, because the entries should always be ordered by the *first* date to appear there, in this case the contents of origdate. In this example, the solution can be as simple as a sortyear field set to something earlier than the date of the other work, e.g., 1951.

And if the reprint dates — in the date field — of the two works were the same? Just as when it is ordering entries, biblatex will always first process the contents of the labelyear field when it is deciding whether to add the extradate alphabetical suffix (a,b,c etc.) to the year to distinguish different works by the same author published in the same year. Our current hypothetical examples would look like this: ([1896] 1974a) and ([1898] 1974b), with the suffixes unnecessary, strictly-speaking, either for ordering or for disambiguating the entries. If the original publication dates — in the origdate field — are the same, and the reprint dates different, you may prefer citations of the two works to read, e.g., (Author [1898a] 1952) and (Author [1898b] 1974), when they in fact read (Author [1898] 1952) and (Author [1898] 1974). These latter forms aren't ambiguous, and even if the reprints themselves appeared in the same year then the alphabetical suffix would appear attached to the date — (Author [1898] 1974a) and (Author [1898] 1974b) — again avoiding ambiguity.

The *Manual* doesn't give clear instructions for how to cope with these situations, but biblatex-chicago-authordate provides help. You can't manually put the alphabetical suffix on an origdate yourself because that field only accepts numerical data. Instead, we

can choose between two solutions. The old way is an unusual expedient, which amounts to switching the two date fields, placing the earlier date in date and the later one in orig-date. The style tests for this condition using a simple arithmetical comparison between the two years, then prints the two dates according to the state of the cmsdate toggle. The three relevant states of this toggle are the same as before, but there are only two possible outcomes, as follows:

1. cmsdate=off (the default) and cmsdate=on *both* print the date at the head of the entry in the list of references and in citations: (Author 1898a), (Author 1898b). As noted above, this style is no longer recommended by the *Manual*, but may still be useful in some cases.

2. cmsdate=both prints both the date and the origdate, using the *Manual's* preferred format: (Author [1898a] 1952), (Author [1898b] 1974). The obsolete options old and new are synonyms for this.

If, for some reason, the automatic switching of the dates cannot be achieved, perhaps in crossref'd letter entries that you really want to have in your list of references (white:ross: memo, white:russ), or perhaps in a reprint edition that hasn't yet appeared in print (preventing the comparison between a year and the word "forthcoming"), then you can use the per-entry option switchdates in the options field to achieve the required effects.

The more drastic method of simplifying the creation of databases with a great many multi-date entries is to use the cmsdate option *in the preamble*. Despite warnings in previous releases, users had plainly already been setting this option in their preambles, so I thought I might at least attempt to make it work as "correctly" as I can. The switches for this option are the same as for the entry-only option, that is:

1. cmsdate=off is the default: (Author 1952).

2. cmsdate=both prints both the origdate and the date, using the *Manual's* standard format: (Author [1898] 1952) in parenthetical citations, Author (1898) 1952 outside parentheses, e.g., in the reference list.

3. cmsdate=on prints the origdate at the head of the entry in the list of references and in citations: (Author 1898). NB: The *Manual* no longer includes this among the approved options. If you want to present the origdate at the head of an entry, then generally speaking you should probably use cmsdate=both. I have nevertheless retained this option for certain cases where it might proved useful. The obsolete options new and old work like both.

The important change for the user is that, when you set this option in your preamble to on or both (or to the obsolete synonyms for the latter, new or old), then biblatex-chicago-authordate will change the default \DeclareLabeldate definition so that the labelyear search order will be origdate, date, eventdate, urldate. This means that for entry types not covered by the avdate option, and for those types as well if you turn off that option, the labelyear will, in any entry containing an origdate, be that very date. If you want *every* such entry to present its origdate in citations and at the head of reference list entries, then setting the option this way makes sense, as you should automatically get the proper extradate and the correct sorting, without having to switch dates around counter-intuitively in your .bib file. A few clarifications may yet be in order.

Obviously, any entry with only a date should behave as usual. Also, since patent entries have fairly specialized needs, I have exempted them from this change to \DeclareLabeldate. Third, the per-entry cmsdate options will still affect which dates are printed in citations and at the head of reference list entries, but they cannot change the search order for the labeldate. This will be fixed by the preamble option. Fourth, if you have been used to switching the date and the origdate to get the correct results, then you should be aware that this mechanism may actually still be useful when using the on switch to cmsdate in the preamble, but it produces incorrect results when the cmsdate option is both in the preamble and the individual entry. The preamble option is designed to make the need for this switching as rare as possible, so some editing of existing databases may be necessary.

Finally, Bertold Schweitzer has brought to my attention certain difficult corner cases involving cross-referenced works with more than one date. In order to facilitate the accurate presentation of such sources, I made a slight change to the way the entry-only `cmsdate=on` and `cmsdate=both` work. If, and only if, a work has only one date, and there is no `switchdates` in the `options` field, then `cmsdate=on` and `cmsdate=both` will both result in the suppression of the `extradate` field in that entry, that is, the year will no longer be printed with its following lowercase letter used to distinguish works by the same `author` published in the same year. Obviously, if the same options are set in the preamble, this behavior is turned off, so that single-date entries will still work properly without manual intervention.

ISO8601-2 Extended Format
Up to this point, the discussion of the `date` field has in fact presented no substantive alterations to the way it behaved in previous releases of `biblatex-chicago`. With this release, however, I have implemented all of the applicable parts of `biblatex`'s elegant, and long standing, support for the ISO8601-2 Extended Format specification, which means the package now provides greatly enhanced possibilities for presenting uncertain and unspecified dates and date ranges, along with date eras, seasons, and time stamps. I have also implemented the *Manual's* (9.64) guidelines for compressing year ranges, as well as providing a few more extras to help with some of the other tricky corners of the *Manual's* instructions. A combination of `biblatex` and `biblatex-chicago` package options allows you to define when, how, and where any of these extended specifications will appear in your documents. I have attempted to provide as compliant a set of defaults as possible in `biblatex-chicago.sty`, but you can alter any of them according to your needs. All are documented in section 5.4, below, but table 3 purports to serve as a convenient reference guide to how this all works.

There are several more general remarks about the `date` field that may be helpful to users of the author-date styles. First, I highly recommend familiarizing yourself with the extended date specifications, as in many cases they will greatly simplify the creation of your .bib databases. A `date` field like `{1957?}` in clark:mesopot not only lets the package provide the appropriate square brackets for you, it also means that the `year` field in the .bbl file sorts just as it should, and can be tested numerically for its relation to other date years in the entry. A `year` field like `{[1957?]}` in the .bib database produces a field in the .bbl that neither sorts nor can be numerically tested. The same holds for a compressed year range, as in tillich:system. With `compressyears` set to `true` by default, a `date` field like `{1951/1963}` lets the package decide what compression is correct, and provides `year` and `endyear` fields that sort and compare numerically for both `switchdates` and `extradate` tests. Clearly, situations may still arise when a specially-crafted `year` or `origyear` field may be necessary, but if you can use the enhanced specifications then I strongly advocate doing so.

verbc
One possible drawback is that using these facilities makes a great many more dates available to the `extradate` mechanism which, it turns out, is something of a mixed blessing. The *Manual* isn't entirely forthcoming about what to do in the (vanishingly rare) case that two works by the same `author` have the same uncertain `date`. `Biblatex-chicago` will print the `extradate` in such situations, so that you could have [1957?]a followed by [ca. 1957]b, which may not be exactly right, nor exactly what you want. Here, the new `verbc` field can help — giving the two entries different values of this field will prevent the `extradate` from appearing. Please see the documentation of the `verbc` field below for all the details.

Second, for most entry types, only a year is really necessary, and in most situations only the year — or year range — will be printed in text citations and at the head of entries in the list of references. More specific `date` fields are often present, however, in an unpredictably broad range of entries. In a change to previous practice, a `date` with a year, month, and day will, even if the year appears at the head of the entry, be repeated in full later in the entry, while if there's only a month (or a season) and a year the month (or season) alone will follow. Also new is the presentation of time stamps, which you can easily provide in your `date` fields (see examples and usage notes in table 3). These finer-grained specifications are really only necessary for news stories that are frequently updated "as they unfold" (14.191), or for online posts, particularly comments, that may need a time stamp for disambiguation. If you wish to specify the time zone, the *Manual* (10.41) prefers

Table 3: Enhanced Date Specifications in biblatex-chicago

Date Specification	Formatted Date (Examples use ᴀᴍᴇʀɪᴄᴀɴ localization)[a]	
	Output Format	**Output Format Notes**
1723?	[1723?][b]	dateuncertain=true set by default
1723~	[ca. 1723][b]	datecirca=true set by default
1723%	[ca. 1723?][b]	Both biblatex options true by default, as above
2016-05-24T15:34:00	May 2, 2016, 3:34 p.m.[c]	alltimes=12h set by default
2016-05-24T15:34:00	May 2, 2016, 15:34[c]	urltime=24h set by default
-0876	877 BC	dateera=christian set in your preamble
-0876/-0866	877–867 BC	
0876	AD 876[d]	dateeraauto=1000 also in preamble
-0876/0866	877 BC–AD 866	
0866/0876	AD 866–76	Cf. compressyears, below
0343-02-03	February 3, AD 343	
-0876	877 BCE	dateera=secular, dateeraauto=1000
-0876/-0866	877–867 BCE	
0876	876 CE	
-0876/0866	877 BCE–866 CE	
0866/0876	866–76 CE	Cf. compressyears, below
0343-02-03	February 3, 343 CE	
195X	1950s[e,f]	Chicago option decaderange=true gets you 1950–59
19XX	20th c.[f]	Chicago option centuryrange=true gets you 1900–1999; alwaysrange=true does the same for this and the previous entry
2004-22	Summer 2004	
1908/1912	1908–12[g]	Chicago option compressyears=true set by default

[a] In other languages both the strings and their placement with respect to the year can and will differ.
[b] The two Chicago options nodatebrackets and noyearbrackets can remove the brackets around the year in this context, though please note that they work quite differently in the notes & bibliography and author-date styles. Please see their documentation in sections 4.4.3 and 5.4.3, respectively.
[c] Any time stamp that is part of a urldate will appear in any entry type, though you can prevent this by setting urlstamp=false. Time stamps in date and origdate fields will appear only in article and periodical entries with a magazine entrysubtype, as well as in all online, review, and suppperiodical entries. Such data in eventdate fields will appear only in review and suppperiodical entries. For timezones the four timezone fields allow you to present Chicago's preferred initialisms ("PST"). The timezones option is true by default. By contrast, the seconds option is not set by default, though you still need to include the seconds, as above, for biber to process the time stamp.
[d] The annodomini string appears before the year *only* in documents in some variant of English.
[e] When the decade string would be ambiguous — "1900s" — the style prints "1900–1909" instead.
[f] For decades and centuries, the current state of the biber code cannot process dates BCE / BC.
[g] The Chicago rules for year-range compression differ from its rules for page-range compression (9.61 & 9.64). Dates BCE / BC are never compressed. You must have loaded biblatex-chicago.sty for the compression code to be available.

initialisms like "EST" or "PDT," and these are most easily provided using the timezone field, where you can include your own parentheses if so desired (cp. 14.191). If you follow the recommendations of the *Manual* and present newspaper and magazine articles "entirely within the text" (15.49), then the citations need to contain the complete date (and
cmsdate=full possible time stamp) along with the journaltitle. Placing cmsdate=full (and skipbib) in the options field of an article or a review entry, alongside a possible useauthor=false, should allow you to achieve this. For online comments in online or review entries this presentation is the default when you use the new commenton relatedtype. (See the documentation of those two types in section 5.1, as well as relatedtype in section 5.2.1.) While we're on this subject, the *Manual* is flexible (in both specifications) on abbreviating the names of months (14.171). By default, biblatex-chicago-authordate uses the full

names, which you can change by setting the option `dateabbrev=true` in the preamble. (Cf. ac:comment, assocpress:gun, barcott:review, batson, creel:house, friends:leia, holiday:fool, nass:address, petroff:impurity, powell:email.)

Third, in the misc entry type the `date` field can help to distinguish between two classes of archival material, letters and "letter-like" sources using `origdate` while others (interviews, wills, contracts) use `date`. (See misc in section 5.1 for the details.) If such an entry, as may well occur, contains only an `origdate`, as can also be the case in letter entries, then Biber and either `\DeclareLabeldate` definition will make it work without further intervention. Fourth, you can in most entry types qualify a `date` with the `userd` field, assuming that the entry contains no `urldate`. For music and video entries, there are several other requirements — please see the documentation of `userd`, below. Fifth, and finally, please note that the `nameaddon` field, which see, is no longer the place for time stamps, as it was in the 16th-edition styles. Any such data there should be moved into the corresponding date field (either the `date` or the `eventdate`, typically).

I recommend that you have a look at dates-test.bib to see how all these complications will affect the construction of your .bib database, especially at the following entries: aristotle:metaphy:gr, creel:house, emerson:nature, james:ambassadors, maitland:canon, maitland:equity, schweitzer:bach, spock:interview, white:ross:memo, and white:russ. Cf. also `origdate`, `timezone`, `verbc`, and `year`, below; the `alldates`, `alltimes`, `alwaysrange`, `centuryrange`, `cmsdate`, `compressyears`, `datecirca`, `dateera`, `dateeraauto`, `dateuncertain`, `decaderange`, `nodatebrackets`, `nodates`, `noyearbrackets`, `switchdates`, `timezones`, `urlstamp`, and `urltime` options in sections 5.4.1, 5.4.3, and 5.4.4; and section 4.5.10 in biblatex.pdf, and section 5.4.1, below, for the `\DeclareLabeldate` command.

day   This field, as of biblatex 0.9, is obsolete, and will be ignored if you use it in your .bib files. Use `date` instead.

doi   Standard biblatex field, providing the Digital Object Identifier of the work. The *Manual* specifies that, given their relative permanence compared to URLs, "authors should prefer a DOI- or Handle-based URL whenever one is available" (14.8). (14.175; friedman:learning). Cf. url.

edition   Standard biblatex field. If you enter a plain cardinal number, biblatex will convert it to an ordinal (chicago:manual), followed by the appropriate string. Any other sort of edition information will be printed as is, though if your data begins with a word (or abbreviation) that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word (or abbreviation) is in lowercase, and biblatex-chicago will automatically do the right thing (babb:peru, times:guide). In most situations, the *Manual* generally recommends the use of abbreviations in the list of references, but there is room for the user's discretion in specific citations (emerson:nature).

editor   As far as possible, I have implemented this field as biblatex's standard styles do, but the requirements specified by the *Manual* present certain complications that need explaining. Biblatex.pdf points out that the editor field will be associated with a title, a booktitle, or a maintitle, depending on the sort of entry. More specifically, biblatex-chicago associates the editor with the most comprehensive of those titles, that is, maintitle if there is one, otherwise booktitle, otherwise title, if the other two are lacking. In a large number of cases, this is exactly the correct behavior (adorno:benj, centinel:letters, plato:republic:gr, among others). Predictably, however, there are numerous cases that require, for example, an additional editor for one part of a collection or for one volume of a multi-volume work. For these cases I have provided the `namea` field. You should format names for this field as you would for author or editor, and these names will always be associated with the title (donne:var).

As you will see below, I have also provided a `nameb` field, which holds the translator of a given title (euripides:orestes). If namea and nameb are the same, biblatex-chicago will concatenate them, just as biblatex already does for editor, translator, and namec (i.e., the compiler). Furthermore, it is conceivable that a given entry will need separate editors for each of the three sorts of title. For this, and for various other tricky situations, there

is the \partedit macro (and its siblings), designed to be used in a note field, in one of the titleaddon fields, or even in a number field (howell:marriage). (Because the strings identifying an editor differ in notes and bibliography, one can't simply write them out in such a field when using the notes & bibliography style, but you can certainly do so in the author-date styles, if you wish. Using the macros will make your .bib file more portable across both Chicago specifications, and also across multiple languages, but they are otherwise unnecessary. Cf. section 7, and also namea, nameb, namec, and translator.)

editora
editorb
editorc
Biblatex provides these fields as a means to specify additional contributors to texts in a number of editorial roles. In the Chicago styles they seem most relevant for the audiovisual types, especially music and video, and now also the performance type, in all of which they help to identify conductors, directors, producers, and performers. To specify the role, use the fields editoratype, editorbtype, and editorctype, which see. (Cf. bernstein:shostakovich, hamilton:miranda, handel:messiah.)

editortype
Normally, with the exception of the article and review types with a magazine entrysubtype, biblatex-chicago-authordate will automatically find a name to put at the head of an entry, starting with an author, and proceeding in order through namea, editor, nameb, translator, and namec (the compiler). If all six are missing, then the title will be placed at the head. (In article and review entries with a magazine entrysubtype, a missing author immediately prompts the use of journaltitle at the head of an entry. See above under article for details.) The editortype field provides even greater flexibility, allowing you to choose from a variety of editorial roles while only using the editor field. You can do this even though an author is named (eliot:pound shows this mechanism in action for a standard editor, rather than for some other role). Two things are necessary for this to happen. First, in the options field you need to set useauthor=false (if there is an author), then you need to put the name you wish to see at the head of your entry into the editor or the namea field. If the "editor" is in fact, e.g., a compiler, then you need to put compiler into the editortype field, and biblatex will print the correct string after the name in the list of references.

In previous releases of biblatex-chicago you could only use defined \bibstrings in this field, at least if you wanted anything printed. N. Andrew Walsh pointed out that the standard biblatex styles will just print the field as-is in this case, allowing them to handle a great many unforeseen editorial roles with comparative ease, so I've implemented this, too, making sure to capitalize the string if the context demands it. The string you choose will differ depending on whether it will be printed after a name at the head of an entry or before a name later on in the entry, e.g., "cartographer" or "maps created by." A bit of trial and error should see you through.

There are a few details of which you need to be aware. Because biblatex-chicago has added the namea field, which gives you the ability to identify the editor specifically of a title as opposed to a maintitle or a booktitle, the editortype mechanism checks first to see whether a namea is defined. If it is, that name will be used at the head of the entry, if it isn't , or if you've set the option usenamea=false, it will go ahead and look for an editor. The editortype field applies only to the editor, but you can use nameatype to modify namea. Biblatex's sorting algorithms, and also its labelname mechanism, should both work properly no matter sort of name you provide, thanks to Biber and the (default) Chicago-specific definitions of \DeclareLabelname and \DeclareSortingTemplate. (Cf. section 5.4.1, below). Please be aware that if you want a shortened form to appear in citations then there's only the shorteditor, which you should ensure presents whichever of the two editors' names (namea or editor) appears at the head of the reference-list entry.

In biblatex 0.9 Lehman reworked the string concatenation mechanism, for reasons he outlines in his RELEASE file, and I have followed his lead. In short, if you define the editortype field, then concatenation is turned off, even if the name of the editor matches, for example, that of the translator. In the absence of an editortype (or nameatype), the usual mechanisms remain in place, that is, if the editor exactly matches a translator and/or a namec, or alternatively if namea exactly matches a nameb and/or a namec, then biblatex will print the appropriate strings. The *Manual* specifically (15.7) recommends not using these

identifying strings in citations, and `biblatex-chicago-authordate` follows that recommendation. If you nevertheless need to provide such a string, you'll have to do it manually in the `shorteditor` field, or perhaps, in a different sort of entry, in a `shortauthor` field.

It may also be worth noting that because of certain requirements in the specification – absence of an `author`, for example – the `useauthor=false` mechanism is either unnecessary or won't work properly in the following entry types: collection, letter, patent, periodical, proceedings, review, suppbook, suppcollection, and suppperiodical.

editoratype
editorbtype
editorctype   These fields identify the exact role of the person named in the corresponding editor[a-c] field, just as `editortype` (q.v.) does for the `editor`. Note that they are not part of the string concatenation mechanism. I have implemented them just as the standard styles do, that is, if the field isn't a pre-defined `\bibstring` it will be printed as-is, contextually capitalized. They have found a use particularly in music, performance, and video entries. Cf. bernstein:shostakovich, hamilton:miranda, and handel:messiah.

eid   Standard `biblatex` field, providing a string or number some journals use uniquely to identify a particular article. Only applicable to the `article` entry type, and only to those without a `magazine` entrysubtype. The 17th edition of the *Manual* now specifies where to print this (14.174), and I have moved it in accordance with its specifications. It replaces the `pages` field in the list of references.

entrysubtype   Standard and very powerful `biblatex` field, left undefined by the standard styles. In `biblatex-chicago-authordate` it has seven very specific uses, the first three of which I have designed in order to maintain, as much as possible, backward compatibility with the standard styles. First, in `article` and `periodical` entries, the field allows you to differentiate between scholarly "journals," on the one hand, and "magazines" and "newspapers" on the other. Usage is fairly simple: you need to put the exact string `magazine` into the `entrysubtype` field if you are citing one of the latter two types of source, whereas if your source is a "journal," then you need do nothing.

The second use involves references to works from classical antiquity and, according to the *Manual*, from the Middle Ages, as well. When you cite such a work using the traditional divisions into books, sections, lines, etc., divisions which are presumed to be the same across all editions, then you need to put the exact string `classical` into the `entrysubtype` field. This has no effect in the list of references, which will still present the particular edition you are using, but it does affect the formatting of in-text citations, in two ways. First, it suppresses some of the punctuation. Second, and more importantly, it suppresses the `date` field in favor of the `title`, so that citations look like (Aristotle *Metaphysics* 3.2.996b5–8) instead of (Aristotle 1997, 3.2.996b5–8). This mechanism may also prove useful in `misc` entries for citations from the Bible or other sacred texts (cf. genesis), and for citing archival collections (house:papers), where it produces citations of the form (House Papers). (Cf. the next but one paragraph.)

If you wish to reference a classical or medieval work by the page numbers of a particular, non-standard edition, then you shouldn't use the `classical` entrysubtype toggle. Also, and the specification is reasonably clear about this, works from the Renaissance and later, even if cited by the traditional divisions, seem to have citations formatted normally, and therefore don't need an `entrysubtype` field. (See *Manual* 14.242–54; aristotle:metaphy:gr, herodotus:wilson, plato:republic:gr; euripides:orestes is an example of a translation cited by page number in a modern edition. Cf. also the notitle option in section 5.4.3.)

The third use of the `entrysubtype` field occurs in `misc` entries. If such an entry contains no such field, then the citation will be treated just as the standard `biblatex` styles would, including the use of italics for the title. Any string at all in `entrysubtype` tells `biblatex-chicago` to treat the source as part of an unpublished archive. Please see section 5.1 above under **misc** for all the details on how these citations work.

Fourth, the field can be defined in the `artwork` entry type in order to refer to a work from antiquity whose title you do not wish to be italicized. Please see the documentation of `artwork` above for the details. Fifth, you can define it in an **audio**, **music**, or **video** entry if such an entry refers to an individual unit that isn't part of any larger collection,

the entry therefore having only a title and not a booktitle, a title that biblatex-chicago would normally interpret as the title of a larger unit (and therefore italicize). Sixth, and sticking with the **video** type, though enacting quite a different syntactic transformation, the 17th edition (14.265) now recommends that, when presenting episodes from a TV series, the name of the series (booktitle) comes before the episode name (title). The exact string tvepisode in the entrysubtype field achieves this reversal, which includes using the booktitle as a sorttitle in the list of references.

Seventh, and finally, you can use any entrysubtype whatever in inreference entries in order to treat them as inherently online works rather than standard published works. See the documentation of **online** and **inreference** entries in section 5.1, above, and also 14.233 and wikiped:bibtex.

eprint
eprintclass
eprinttype Kazuo Teramoto suggested adding biblatex's excellent eprint handling to biblatex-chicago, and he sent me a patch implementing it. I have applied it, with minor alterations, so these three fields work more or less as they do in standard biblatex. They may prove helpful in providing more abbreviated references to online content than conventional URLs, though I can find no specific reference to them in the *Manual*.

eventdate This is a standard biblatex field which has gradually accumulated functions in biblatex-chicago. It can now play a role in artwork, audio, image, inproceedings, music, proceedings, review, standard, suppperiodical, unpublished, and video entries. In artwork and image entries it identifies the publication date of, most frequently, a photograph, in association with the howpublished field which identifies the periodical or other medium in which it was published (mccurry:afghangirl). In standard entries it will also usually be associated with a howpublished field, allowing you to specify a later renewal or reaffirmation of a standard (niso:bibref). In audio entries, it specifies the release date of a single episode of a podcast (danforth:podcast). In music entries, it identifies the recording or performance date of a particular song (rather than of a whole disc, for which you would use origdate), whereas in video entries it identifies either the original broadcast date of a particular episode of a TV series or the date of a filmed musical performance. In both these cases biblatex-chicago will automatically prepend a bibstring — recorded and aired, respectively — to the date, but you can change this string using the userd field, something you'll definitely want to do for filmed musical performances (friends:leia, handel:messiah, holiday:fool).

In inproceedings, proceedings, and unpublished entries it identifies the date of an event at which a published or unpublished work was presented, though in truth the date will do as well in unpublished entries (nass:address). The field's use in review and suppperiodical entries, finally, includes a possible time stamp. In this context, an eventdate helps to identify a particular comment on, or reply to another comment on, a blog post. Given that many such posts by a single author could appear on the same day, you can distinguish them by putting a time specification in the eventdate field itself (ac:comment). Please see the **review** type, above, for the details of how to cite these materials, possibly with the help of the new commenton relatedtype. See also the date field docs above, in particular table 3, for details on how the ISO8601-2 Extended Format specifications offered by biblatex, including time stamps and much else besides, have been implemented in biblatex-chicago.

In the default configuration of \DeclareLabeldate, dates for citations and for the head of reference list entries are searched for in the order date, eventdate, origdate, urldate. This suits the Chicago author-date styles very well, except for music, review, standard, suppperiodical, and video entries. In music and video entries the general rule is to emphasize the earliest date, whether that be, for example, the recording date or original release date (15.57). The other three types have special requirements that once again necessitate putting the eventdate at the head of the queue. For these five entry types, then, \DeclareLabeldate uses the order eventdate, origdate, date, urldate. (See the avdate option in section 5.4.2, below.)

eventtimezone This field can, if necessary, specify the time zone associated with a time stamp given as part of an eventdate. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

eventtitle   A standard biblatex field for identifying the name of the event that produces either a published record (inproceedings and proceedings entries) or an unpublished one (unpublished).

eventtitleaddon   Standard biblatex field for adding information about an eventtitle, and available in the same entry types as that field.

foreword   As with the afterword field above, foreword will in general function as it does in standard biblatex. Like afterword (and introduction), however, it has a special meaning in a suppbook entry, where you simply need to define it somehow (and leave afterword and introduction undefined) to make a foreword the focus of a citation.

holder   A standard biblatex field for identifying a patent's holder(s), if they differ from the author. The *Manual* has nothing to say on the subject, but biblatex-chicago prints it (them), in parentheses, just after the author(s).

howpublished   Standard biblatex field which, like the eventdate field, is gradually accumulating functions in biblatex-chicago. In the booklet type it retains something of its traditional usage, replacing the publisher, and has a similar (somewhat paradoxical) place in unpublished entries. In the misc and performance types it works almost as a second note field, bringing in extra information about a work in close association with the type and version fields, while the dataset entry type associates it both with those two fields and with the number field. 17th-edition music entries require a field to provide the medium of downloaded music and/or the name of the streaming service, so howpublished works there as an online double of type and of publisher. Finally, in artwork, image, and standard entries it serves to qualify or modify an eventdate, almost as a userd field modifies a date or urldate. Please see the docs of those entry types for more information, and also bedford:photo, clark:mesopot, mccurry:afghangirl, niso:bibref, rihanna:umbrella.

institution   Standard biblatex field. In the thesis entry type, it will usually identify the university for which the thesis was written, while in a report entry it may identify any sort of institution issuing the report.

introduction   As with the afterword and foreword fields above, introduction will in general function as it does in standard biblatex. Like those fields, however, it has a special meaning in a suppbook entry, where you simply need to define it somehow (and leave afterword and foreword undefined) to make an introduction the focus of a citation.

isbn   Standard biblatex field, for providing the International Standard Book Number of a publication. Not typically required by the *Manual*.

isrn   Standard biblatex field, for providing the International Standard Technical Report Number of a report. Only relevant to the report entry type, and not typically required by the *Manual*.

issn   Standard biblatex field, for providing the International Standard Serial Number of a periodical in an article or a periodical entry. Not typically required by the *Manual*.

issue   Standard biblatex field, designed for article or periodical entries identified by something like "Spring" or "Summer" rather than by the usual month or number fields (brown:bremer). Biblatex's enhanced date handling allows you to specify a season in the date field, with the "months" 21–24 used for Spring, Summer, Autumn, and Winter, respectively. Cf. table 3, above.

issuesubtitle   The subtitle for an issuetitle — see next entry.

issuetitle   Standard biblatex field, intended to contain the title of a special issue of any sort of periodical. If the reference is to one article within the special issue, then this field should be used in an article entry (conley:fifthgrade), whereas if you are citing the entire issue as a whole, then it would go in a periodical entry, instead (good:wholeissue). The note field

is the proper place to identify the type of issue, e.g., `special issue`, with the initial letter lower-cased to enable automatic contextual capitalization.

journalsubtitle   The subtitle for a journaltitle — see next entry.

journaltitle   Standard biblatex field, replacing the standard BIBTEX field journal, which, however, still works as an alias. It contains the name of any sort of periodical publication, and is found in the article and review entry types. In the case where a piece in an article or review (entrysubtype magazine) doesn't have an author, biblatex-chicago provides for this field to be used as the author. See above (section 5.2) under **article** for details. The lakeforester:pushcarts and nyt:trevorobit entries in dates-test.bib will give you some idea of how this works. Please note there is a shortjournal field which you can use to abbreviate the journaltitle in citations and/or in the reference list, and you can also use it to print a list of journal abbreviations. Cf. the shortjournal documentation below.

journaltitleaddon   An annex to the journaltitle, for which see previous entry. Such an annex would be printed in the main text font. If your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago-authordate will automatically do the right thing. The package and entry option jtitleaddon (section 5.4.2) allows you to customize the punctuation that appears before the journaltitleaddon field (hua:cms). The default is a `space`.

keywords   This field is biblatex's powerful and flexible technique for filtering entries in a list of references, allowing you to subdivide it according to just about any criteria you care to invent, or indeed to prevent entries in citations from appearing in reference list, as the *Manual* sometimes recommends. See biblatex.pdf (3.7) for thorough documentation.

language   A standard biblatex field, designed to allow you to specify the language(s) in which a work is written. As a general rule, the Chicago style doesn't require you to provide this information, though it may well be useful for clarifying the nature of certain works, such as bilingual editions, for example. There is at least one situation, however, when the *Manual* does specify this data, and that is when the title of a work is given in translation, even though no translation of the work has been published, something that might happen when a title is in a language deemed to be unparseable by a majority of your expected readership (14.99; chu:panda, pirumova, rozner:liberation). In such a case, you should provide the language(s) involved using this field, connecting multiple languages using the keyword `and`. (I have retained biblatex's `\bibstring` mechanism here, which means that you can use the standard bibstrings or, if one doesn't exist for the language you need, just give the name of the language, capitalized as it should appear in your text. You can also mix these two modes inside one entry without apparent harm.)

An alternative arrangement suggested by the *Manual* is to retain the original title of a piece but then to provide its translation, as well. If you choose this option, you'll need to make use of the **usere** field, on which see below. In effect, you'll probably only ever need to use one of these two fields in any given entry, and in fact biblatex-chicago will only print one of them if both are present, preferring usere over language for this purpose (see kern, pirumova:russian, and weresz). Note also that both of these fields are universally associated with the title of a work, rather than with a booktitle or a maintitle. If you need to attach a language or a translation to either of the latter two, you could probably manage it with special formatting inside those fields themselves.

lista   I intend this field specifically for presenting citations from reference works that are arranged alphabetically, where the name of the article rather than a page or volume number should be given. The field is a biblatex list, which means you should separate multiple items with the keyword `and`. Each item receives its own set of quotation marks, and the whole list will be prefixed by the appropriate string ("s.v.," *sub verbo*, pl. "s.vv."). Biblatex-chicago will only print such a field in a book or an inreference entry, and you should look at the documentation of these entry types for further details. (See *Manual* 14.232–33; grove:sibelius, times:guide, wikiped:bibtex.)

location This is biblatex's version of the usual BIBTEX field address, though the latter is accepted as an alias if that simplifies the modification of older .bib files. According to the *Manual* (14.129), a citation usually need only provide the first city listed on any title page, though a list of cities separated by the keyword "and" will be formatted appropriately. If the place of publication is unknown, you can use \autocap{n}.p. instead (14.132). For all cities, you should use the common English version of the name, if such exists (14.131).

Two other uses need explanation here. In article, periodical, and review entries, there is usually no need for a location field, but "if a journal might be confused with another with a similar title, or if it might not be known to the users of a bibliography," then this field can present the place or institution where it is published (14.182, 14.191, 14.193–94; garrett, kimluu:diethyl, and lakeforester:pushcarts). For blogs cited using article entries, this is a good place to identify the nature of the source — i.e., the word "blog" — letting the style automatically provide the parentheses (15.51; ellis:blog).

mainsubtitle The subtitle for a maintitle — see next entry.

maintitle The main title for a multi-volume work, e.g., "Opera" or "Collected Works." It no longer takes sentence-style capitalization in authordate, though it does in authordate-trad. In cross references produced using the crossref field, the title of **mv*** entry types always becomes a maintitle in the child entry. (See donne:var, euripides:orestes, harley:cartography, lach:asia, pelikan:christian, and plato:republic:gr.)

Because the 17th edition of the *Manual* recommends that you present not only the names of blogs but also the names of their parent (usually periodical) publications, I have added this field to article, periodical, and review entries for just this purpose. See the documentation of those entry types in section 5.1, above, and also table 2 (15.51; amlen:hoot).

maintitleaddon An annex to the maintitle, for which see previous entry. Such an annex would be printed in the main text font. If your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago will automatically do the right thing. The package and entry options ptitleaddon and ctitleaddon (section 5.4.2) allow you to customize the punctuation that appears before the maintitleaddon field (schubert:muellerin).

month Standard biblatex field, containing the month of publication. This should be an integer, i.e., month={3} not month={March}. See date for more information.

namea This is one of the fields biblatex provides for style writers to use, but which it leaves undefined itself. In biblatex-chicago it contains the name(s) of the editor(s) of a title, if the entry has a booktitle and/or a maintitle, in which situation the editor would be associated with one of these latter fields (donne:var). (In article and review entries, namea applies to the title instead of the issuetitle, should the latter be present.) You should present names in this field exactly as you would those in an author or editor field, and the package will concatenate this field with nameb if they are identical. When choosing a name for a citation or to head a reference-list entry, biblatex-chicago gives precedence to namea over editor. See under **editor** and **editortype** above for the full details. Please note that, as the field is highly single-entry specific, namea isn't inherited from a crossref'ed parent entry. Please note, also, that you can use the nameatype field to redefine this role just as you can with editortype, which see. Cf. also nameb, namec, translator, and the macros \partedit, \parttrans, \parteditandtrans, \partcomp, \parteditandcomp, \parttransandcomp, and \parteditttransandcomp, for which see section 5.3.1.

nameaddon This field is provided by biblatex, though not used by the standard styles. In biblatex-chicago its primary use, in most entry types, has always been to specify that an author's name is a pseudonym, or to provide either the real name or the pseudonym itself, if the other is being provided in the author field. The abbreviation "pseud." (always lowercase in English) is specified, either on its own or after the pseudonym (centinel:letters, creasey:ashe:blast, creasey:morton:hide, creasey:york:death, and lecarre:quest); remember that \bibstring{pseudonym} does the work for you. See under **author** above for the full details.

The field has slowly accumulated other functions, so when Philipp Immel made a feature request, and pointed to a discussion on Stack Exchange which suggested a few more, I thought I might generalize the field's functionality, providing three package options to allow users to mould it to their needs. Before discussing these, allow me to emphasize that the package defaults remain exactly the same as before, so that, absent any of the new options, the style still provides square brackets around the nameaddon in most entry types, no brackets of any sort in online, review, and suppperiodical entries, as well as in misc entries with an entrysubtype, and rather specialized handling in customc entries (which ignore all but the first of the new options — see below). If you're happy with the status quo, then no changes to your documents or .bib databases are necessary.

nameaddon   If you do need or want to put the field to a different use, the following options may help. All of them are available globally, per type, and per entry. The first new option is simply called nameaddon, and determines where and when the field will be printed at all. There are three possible values:

> **all:** This is the default; if an entry has a nameaddon, it will appear in the reference list.
> **none:** The field will not appear in the reference list.
> **first:** Philipp Immel requested this as a way to provide an author's dates in the nameaddon field and only have them printed the first time that author appears in the reference list. The code tests for identical nameaddon fields in works by identical authors, so other sorts of nameaddon will be printed as usual.

nameaddonsep   The nameaddonsep option controls the punctuation that appears before the nameaddon. It takes the following six keys:

> **space** = \addspace. This is the default.
> **none** = no separator at all. It presumes that you will include one in the nameaddon field itself.
> **colon** = \addcolon\addspace.
> **comma** = \addcomma\addspace.
> **period** = \addperiod\addspace.
> **semicolon** = \addsemicolon\addspace.

nameaddon-   The nameaddonformat option allows you to change the format of the nameaddon field
format   on the fly, so its value should be a field format that biblatex understands. This includes standard formats like parens, brackets or emph, and also custom formats that you provide in your preamble using \DeclareFieldFormat, in case the standard ones aren't adequate. If you don't define this option, then the usual defaults apply, as delineated above, and you can use your own parentheses in online, review, and suppperiodical entries, as well as in misc entries with an entrysubtype, to distinguish screen names or other authorial information from traditional pseudonyms (in brackets).

There are two more details. If you are using the 17th-edition styles for the first time, please note that the 16th edition of the *Manual* recommended specifying comments to blogs and other online content using a time stamp in parentheses after the author, but the 17th edition handles time stamps both differently and more widely, so in this case you would now put time data into the date or eventdate field, particularly when the date itself is too coarse a specification to identify a comment unambiguously. Also, the *Manual* (15.52) now specifies that comments should appear "only in the text, in reference to the related post," so I've provided some new functionality to enable this. Please see the **online** and **review** types, above, especially table 2, for the details of how to cite these materials, possibly with the help of the new commenton relatedtype and a separate **customc** entry. See also the date field docs above, in particular table 3, for details on how the ISO8601-2 Extended Format specifications offered by biblatex, including time stamps and much else besides, have been implemented in biblatex-chicago. (Cf. ellis:blog, obrien:recycle).

In the customc entry type, finally, which is used to create alphabetized cross-references to other bibliography entries, the nameaddon field allows you to change the default string linking the two parts of the cross-reference. The code automatically tests for a known bibstring, which it will italicize. Otherwise, it prints the string as you've provided in the nameaddon field itself. The punctuation is fixed.

nameatype You can use this field to change the role of a namea just as you can use editortype to change the role of an editor. As with the editortype, using this field prevents string concatenation with identical nameb or namec fields. Please see **editortype**, above, for the details.

nameb Like namea, above, this is a field left undefined by the standard biblatex styles. In biblatex-chicago, it contains the name(s) of the translator(s) of a title, if the entry has a booktitle or maintitle, or both, in which situation the translator would be associated with one of these latter fields (euripides:orestes). (In article and review entries, nameb applies to the title instead of the issuetitle, should the latter be present.) You should present names in this field exactly as you would those in an author or translator field, and the package will concatenate this field with namea if they are identical. See under the **translator** field below for the full details. Please note that, as the field is highly single-entry specific, nameb isn't inherited from a crossref'ed parent entry. Please note, also, that in biblatex-chicago's name-finding algorithms nameb takes precedence over translator. Cf. also namea, namec, origlanguage, translator, userf and the macros \partedit, \parttrans, \parteditand-trans, \partcomp, \parteditandcomp, \parttransandcomp, and \partedittransand-comp in section 5.3.1.

namec The *Manual* (15.36) specifies that works without an author may be listed under an editor, translator, or compiler, assuming that one is available, and it also specifies the strings to be used with the name(s) of compiler(s). All this suggests that the *Manual* considers this to be standard information that should be made available in a bibliographic reference, so I have added that possibility to the many that biblatex already provides, such as the editor, translator, commentator, annotator, and redactor, along with writers of an introduction, foreword, or afterword. Since biblatex doesn't offer a compiler field, I have adopted for this purpose the otherwise unused field namec. It is important to understand that, despite the analogous name, this field does not function like namea or nameb, but rather like editor or translator, and therefore if used will be associated with whichever title field these latter two would be were they present in the same entry. Identical fields among these three will be concatenated by the package, and concatenated too with the (usually) unnecessary commentator, annotator and the rest. Also please note that I've arranged the concatenation algorithms to include namec in the same test as namea and nameb, so in this particular circumstance you can, if needed, make namec analogous to these two latter, title-only fields. (See above under **editortype** for details of how you can use that field, or the nameatype field, to identify a compiler.)

It might conceivably be necessary at some point to identify the compiler(s) of a title separate from the compiler(s) of a booktitle or maintitle, but for the moment I've run out of available name fields, so you'll have to fall back on the \partcomp macro or the related \parteditandcomp, \parttransandcomp, and \partedittransandcomp, on which see Commands (section 5.3.1) below. (Future releases may be able to remedy this.) It may be as well to mention here too that of the three names that can be substituted for the missing author at the head of an entry, biblatex-chicago will choose a namea if present, then an editor, a nameb, or a translator, with namec coming last, assuming that the fields aren't identical, and therefore to be concatenated. Biblatex's sorting algorithms, and also its labelname mechanism, should both work properly no matter what sort of name you provide, but do please remember that if you want the package to skip over any names you can employ the use<name>=false options. Indeed, biblatex's usenamec has replaced the old Chicago-specific usecompiler, which is deprecated.

note As in standard biblatex, this field allows you to provide bibliographic data that doesn't easily fit into any other field. In this sense, it's very like addendum, but the information provided here will be printed just before the publication data. (See chaucer:alt, cook:sotweed, emerson:nature, and rodman:walk for examples of this usage in action.) It also has a specialized use in the periodical types (article, periodical, and review), where it holds supplemental information about a journaltitle, such as "special issue" (conley:fifth-grade, good:wholeissue). In all uses, if your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, then simply ensure that that word is in lowercase, and biblatex-chicago will automatically do the right thing. Cf. addendum.

number   This is a standard biblatex field, steadily accumulating uses in biblatex-chicago. It may contain the number of a journaltitle in an article or review entry, the number of a title in a periodical entry, the volume/number of a book (or musical recording) in a series, the (generally numerical) specifier of the type in a report entry, the archive location (or database accession number) of a dataset entry, and the number of a national or international standard in a standard entry. Generally, in an article, periodical, or review entry, this will be a plain cardinal number, but in such entries biblatex-chicago does the right thing if you have a list or range of numbers (unsigned:ranke). In any book-like entry it may well contain considerably more information, including even a reference to "2nd ser.," for example, while the series field in such an entry will contain the name of the series, rather than a number. This field is also the place for the patent number in a patent entry. Cf. issue and series. (See 14.123–25 and boxer:china, palmatary:pottery, wauchope:ceramics; 14.171 and beattie:crime, conley:fifthgrade, friedman:learning, garrett, gibbard, hlatky:hrt, mcmillen:antebellum, rozner:liberation, warr:ellison; 14.257 and genbank:db; 14.259 and niso:bibref; 14.263 and holiday:fool.)

NB: This may be an opportune place to point out that the *Manual* (14.147) prefers arabic to roman numerals in most circumstances (chapters, volumes, series numbers, etc.), even when such numbers might be roman in the work cited. The obvious exception is page numbers, in which roman numerals indicate that the citation came from the front matter, and should therefore be retained.

options   A standard biblatex field, for setting certain options on a per-entry basis rather than globally. Information about some of the more common options may be found above under author and date, and below in section 5.4.3. See creel:house, eliot:pound, emerson:nature, ency:britannica, herwign:office, lecarre:quest, and maitland:canon for examples of the field in use.

organization   A standard biblatex field, retained mainly for use in the misc, online, and manual entry types, where it may be of use to specify a publishing body that might not easily fit in other categories. In biblatex, it is also used to identify the organization sponsoring a conference in a proceedings or inproceedings entry, and I have retained this as a possibility, though the *Manual* is silent on the matter.

origdate   This is a standard biblatex field which allows more than one full date specification for those references which need it. (You can also provide a time stamp in the field, after an uppercase "T", but I foresee this being very rarely needed in the author-date styles. See table 3 for biblatex-chicago's implementation of biblatex's enhanced date specifications.) As with the analogous date field, you provide the date (or range of dates) in ISO8601 format, i.e., yyyy-mm-dd. In most entry types, you would use origdate to provide the date of first publication of a work, most usually needed only in the case of reprint editions, but also recommended by the *Manual* for electronic editions of older works (15.40, 14.114, 14.162; aristotle:metaphy:gr, emerson:nature, james:ambassadors, schweitzer:bach). In both the letter and misc (with entrysubtype) entry types, the origdate identifies when a letter (or similar) was written. In such misc entries, some "non-letter-like" materials (like interviews) need the date field for this purpose, while in letter entries the date applies to the publication of the whole collection. If such a published collection were itself a reprint, judicious use of the pubstate field or perhaps improvisation in the location field might be able to rescue the situation. (See white:ross:memo, white:russ, and white:total for how letter entries can work; creel:house shows the field in action in a misc entry, while spock:interview uses date instead.)

Because of the importance of date specifications in the author-date styles, biblatex-chicago-authordate and authordate-trad provide options and automated behaviors that allow you to emphasize the origdate in citations and at the head of entries in the list of references. In entries which have *only* an origdate — usually misc with an entrysubtype — Biber and the default \DeclareLabeldate configuration make it possible to do without a cmsdate option, as the origdate will automatically appear where and as it should. In book-like entries with both a date and an origdate, the *Manual* recommends that you present, in citations and at the head of reference list entries, only the date or both dates together. The

latter is accomplished using the cmsdate entry option. In some cases it may even be necessary to reverse the two date fields, putting the earlier year in date and the later in origdate. If your reference apparatus contains many such instances, it may well be convenient for you instead to use the cmsdate preamble option, which I have designed in an attempt to reduce the amount of manual intervention needed to present lots of entries with multiple dates. In short, setting cmsdate to both or on in the preamble promotes the origdate to the top of the search for a labeldate to use in citations and at the head of entries in the reference list. This can solve many problems with the extradate field — 1978**a** — and also with sorting in the reference list. Please see above under **date** for all the details on how these options interact.

cmsdate
in preamble

In the default configuration of \DeclareLabeldate, dates for citations and for the head of reference list entries are searched for in the order date, eventdate, origdate, urldate. If you set the cmsdate preamble options I've just mentioned, this changes to origdate, date, eventdate, urldate. These generally cover the needs of the Chicago author-date styles well, except for music, standard, and video entries, and, exceptionally, some review and suppperiodical entries. Here the general rule is to emphasize the earliest date. For these five entry types, then, \DeclareLabeldate uses the order eventdate, origdate, date, urldate. In music entries, you can use the origdate in two separate but related ways. First, it can identify the recording date of an entire disc, rather than of one track on that disc, which would go in eventdate. (Compare holiday:fool with nytrumpet:art.) Second, the origdate can provide the original release date of an album. For this to happen, you need to put the string reprint in the pubstate field, which is the standard mechanism across many other entry types for identifying a reprinted work. (See floyd:atom.) In video entries, the origdate is intended for the original release date of a film, whereas the eventdate would hold the original broadcast date of, e.g., an episode of a TV series. In both these two entry types, the style will, depending on the context, automatically prepend appropriate bibstrings to the origdate. You can, assuming you've not activated the pubstate mechanism in a music entry, choose a different string using the userd field, but please be aware that if an entry also has an eventdate, then userd will apply to that, instead, and you'll be forced to accept the default string. (Compare friends:leia with hitchcock:nbynw; 15.57, 14.263–65; Cf. cmsdate in sections 5.4.3 and 5.4.4, \DeclareLabeldate in section 5.4.1, and avdate in section 5.4.2.)

A couple of further notes are in order. First, artwork and image entries (which see) have their own scheme, and are not governed by the avdate option. Here, the style uses the earlier of two dates as the creation date of the work while the later is the printing date of, e.g., a particular exemplar of a photograph or of an etching. Depending on how you want this information presented in an entry, you can distribute these dates between the date and origdate fields as you wish. Second, because the origdate field only accepts numbers, some improvisation may be needed if you wish to include "n.d." (\bibstring{nodate}) in an entry. In letter and misc, this information can be placed in titleaddon, but in other entry types you may need to use the location field. (The origyear field usually works, too.)

origlanguage
origlocation See section 5.2.1, below.
origpublisher

origtimezone This field can, if necessary, specify the time zone associated with a time stamp given as part of an origdate. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

pages This is the standard biblatex field for providing page references. In many article entries you'll find this contains something other than a page number, e.g. a section name or edition specification (14.191; kozinn:review, nyt:trevorobit). Of course, the same may be true of almost any sort of entry, though perhaps with less frequency. Curious readers may wish to look at brown:bremer (14.180) for an example of a pages field used to facilitate reference to a two-part journal article. Cf. number for more information on the *Manual*'s preferences regarding the formatting of numerals; bookpagination and pagination provide details about biblatex's mechanisms for specifying what sort of division a given pages field contains; and usera discusses a different way to present the section information pertaining to a newspaper article.

David Gohlke brought to my attention a discussion that took place a couple of years ago on Stackexchange regarding the automatic compression of page ranges, e.g., 101--109 in the .bib file or in the postnote field would become 101–9 in the document. Biblatex has long had the facilities for providing this, and though the *Manual's* rules (9.61) are fairly complicated, Audrey Boruvka fortunately provided in that discussion code that implements the specifications. As some users may well be accustomed to compressing page ranges themselves in their .bib files, and in their postnote fields, I have made the activation of this code a package option, so setting compresspages=true when loading biblatex-chicago should automatically give you the Chicago-recommended page ranges. **NB**: the code now resides in biblatex-chicago.sty, so if you don't load that package then you'll need to copy the code into your preamble for the option to have the desired effect.

pagination This, a standard biblatex field, allows you automatically to prefix the appropriate identifying string to information you provide in the postnote field of a citation command, whereas bookpagination allows you to prefix a string to the pages field. Please see **bookpagination** above for all the details on this functionality, as aside from the difference just mentioned the two fields are equivalent.

part Standard biblatex field, which identifies physical parts of a single logical volume in booklike entries, not in periodicals. It has the same purpose in biblatex-chicago, but because the *Manual* (14.121) calls such a thing a "book" and not a "part," the string printed in the list of references will, at least in English, be "bk." instead of the plain dot between volume number and part number (harley:cartography, lach:asia). If the field contains something other than a number, biblatex-chicago will print it as is, capitalizing it if necessary, rather than supplying the usual bibstring, so this provides a mechanism for altering the string to your liking. The field will be printed in the same place in any entry as would a volume number, and although it will most usually be associated with such a number, it can also function independently, allowing you to identify parts of works that don't fit into the standard scheme. If you need to identify "parts" or "books" that are part of a published series, for example, then you'll need to use a different field, (which in the case of a series would be number [palmatary:pottery]). Cf. volume; iso:electrodoc.

publisher Standard biblatex field. Remember that "and" is a keyword for connecting multiple publishers, so if a publisher's name contains "and," then you should either use the ampersand (&) or enclose the whole name in additional braces. (See *Manual* 14.133–41; aristotle:metaphy:gr, cohen:schiff, creasey:ashe:blast, dunn:revolutions.)

There are, as one might expect, a few further subtleties involved here. If you give two publishers in the field they will both be printed, separated by a forward slash in both notes and bibliography (14.90; sereny:cries). The 17th edition generally is rather keener than the 16th on using just one, particularly so in the case when the parent company of an imprint is also listed on a title page, in which case only the imprint need be included in your apparatus (14.138). If an academic publisher issues "certain books through a special publishing division or under a special imprint or as part of a publishing consortium (or joint imprint)," this arrangement may be specified in the publisher field (14.139; cohen:schiff). If a book has two co-publishers "in different countries" (14.140), then the simplest thing to do is to choose one, probably the nearest one geographically. If you feel it necessary to include both, then levistrauss:savage demonstrates one way of doing so, using a combination of the publisher and location fields. If the work is self-published, you can specify this in the pubstate field (see below), and any commercial self-publishing platform would go in publisher (14.137). Books published before 1900 can, at your discretion, include only the place (if known) and the date (14.128). If for some reason you need to indicate the absence of a publisher, the abbreviation given by the *Manual* is n.p., though this can also stand for "no place." The *Manual* also mentions s.n. (= *sine nomine*) to specify the lack of a publisher (10.42).

pubstate In response to new specifications in the 17th edition of the *Manual* (esp. 14.137), I have tried to generalize the functioning of the pubstate field in all entry types. Because the author-date style has fairly complicated rules about presenting reprinted editions (15.40), the reprint string still has a special status. Depending on which date(s) you have chosen

to appear at the head of the entry, biblatex-chicago-authordate will either print the (localized) string reprint in the proper place or otherwise provide a notice at the end of the entry detailing the original publication date. See under **date** above for the available permutations. (Cf. aristotle:metaphy:gr, maitland:canon, maitland:equity, schweitzer:bach.)

NB Other strings are divided into two types: those which biblatex-chicago will print as the year, which currently means *only* those for which biblatex contains bibstrings indicating works soon to be published, i.e., forthcoming, inpreparation, inpress, and submitted; and those, i.e., everything else, which will be printed before, and in close association with, other information about the publisher of a work. (This is a change from previous behavior, where non-reprint strings were printed *after* the publication information, as in the standard styles. You can still use the addendum field to present information here, of course.) The four strings that replace the year will always be localized, as will reprint and selfpublished (and anything else that biblatex finds to be a \bibstring) from the second category. All other strings will be printed as-is, capitalized if needed, just before the publisher (author:forthcoming, contrib:contrib, schweitzer:bach).

There is one further subtlety of which you ought to be aware. In music and video entries, the reprint string in pubstate will only make a difference to your entries when the date which it modifies — the origdate, typically — *doesn't* appear in citations and at the head of reference-list entries. In this case the date is treated as an original release date, and it will be printed, preceded by the appropriate string, near the end of the entry. Other strings don't show this special behavior in these entries.

redactor I have implemented this field just as biblatex's standard styles do, even though the *Manual* doesn't actually mention it. It may be useful for some purposes. Cf. annotator and commentator.

reprinttitle See section 5.2.1, below.

series A standard biblatex field, usually just a number in an article, periodical, or review entry, almost always the name of a publication series in book-like entries, and providing similar identifying information associated with a number in music and standard entries. If you need to attach further information to the series name in a book-like entry, then the number field is the place for it, whether it be a volume, a number, or even something like "2nd ser." or "\bibstring{oldseries}." Of course, you can also use \bibstring{oldseries} or \bibstring{newseries} in an article entry, but there you would place it in the series field itself. (In fact, the series field in article and periodical entries is one of the places where biblatex allows you just to use the plain bibstring oldseries, for example, rather than making you type \bibstring{oldseries}. The type field in manual, patent, report, and thesis entries also has this auto-detection mechanism in place; see the discussion of \bibstring below for details.) In whatever entry type, these bibstrings produce the required abbreviation. (For books and similar entries, see *Manual* 14.123–26; boxer:china, browning:aurora, palmatary:pottery, plato:republic:gr, wauchope:ceramics; for periodicals, see 14.184; garaud:gatine, sewall:letter.) Cf. number for more information on the *Manual*'s preferences regarding the formatting of numerals.

shortauthor This is a standard biblatex field, but biblatex-chicago makes considerably greater use of it than the standard styles. For the purposes of the author-date specification, the field provides the name to be used in text citations. In the vast majority of cases, you don't need to specify it, because the biblatex system selects the author's last name from the author field and uses it in such a reference, and if there is no author it will search namea, editor, nameb, translator, and namec, in that order. The current versions of biblatex and Biber will automatically alphabetize by any of these names if they appear at the head of an entry. If, in an author-less article entry (entrysubtype magazine), you allow biblatex-chicago to use the journaltitle as the author — the default behavior — and you have been accustomed to using the shortauthor field to abbreviate it, it may be simpler now to use the shortjournal field instead, which does all of the formatting for you, and additionally adds the possibility of printing a list of journal abbreviations. See just below for the details. (Cf. gourmet:052006, lakeforester:pushcarts, nyt:trevorobit, unsigned:ranke). With long,

institutional authors, a shortened version in shortauthor may save space in the running text (evanston:library), but see under **shorthand** for another method of saving space.

As mentioned under editortype, the *Manual* (15.36) recommends against providing the identifying string (e.g., ed. or trans.) in text citations, and biblatex-chicago follows their recommendation. If you need to provide these strings in such a citation, then you'll have to do so by hand in the shortauthor field, or in the shorteditor field, whichever you are using.

shorteditor   Like shortauthor, a field to provide a name for a text citation, in this case for, e.g., a collection entry that typically lacks an author. The shortauthor field works just as well in most situations, but if you have set useauthor=false (and not useeditor=false) in an entry's options field, then only shorteditor will be recognized. It may be worth pointing out that, because biblatex-chicago also provides a namea field for the editor of a title as opposed to a main- or booktitle, and because in standard use the namea, if present, will be chosen to head a reference list entry before the editor, you should present the shortened namea here instead of a shortened editor in such cases. Cf. editortype, above.

shorthand   This is biblatex's mechanism for using abbreviations in citations. For biblatex-chicago-authordate I have modified it somewhat to conform to the needs of the specification, though there is a package option to revert the behavior to something closer to the biblatex standard — see below and under cmslos in section 5.4.2. The main problem when presenting readers with an abbreviation is to ensure that they know how to expand it. In the notes & bibliography style this is accomplished with a notice in the first footnote citing a given work, which explains that henceforth the abbreviation will be used instead, and also, if needed, with a list of shorthands that summarizes all the abbreviations used in a particular text. The first part of this system isn't available in the author-date style of citation, and indeed these citations are in themselves already highly-abbreviated keys to the fuller information to be found in the list of references. There are cases, however, particularly when institutions or journaltitles appear as authors, when you may feel the need to provide a shortened version for citations. I have already discussed two options available to you just above (cf. **shortauthor** and **shortjournal**). For the former to work the abbreviation must either be instantly recognizable to your readership or at least easily parseable by them, while with the latter you can either rely on the conventions of your field or, alternately, provide a list of journal abbreviations using \printbiblist{shortjournal}.

For long institutional names the *Manual's* recommendation (15.37) involves using an abbreviation which will appear not only in citations but also at the head of the entry in the list of references. Such an entry should therefore be alphabetized by the abbreviation, with its expansion placed (inside parentheses) between the abbreviation and the date. This formatting can be produced in one of two ways: either you can provide a specially-formatted author field (for the reference list, and including both the abbreviation and the parenthesized expansion) + a shortauthor (for the citations), or you can use a normal author field + a shorthand, in which case biblatex-chicago-authordate will automatically use the shorthand in text citations and also place it at the head of the reference list entry, followed by the author within parentheses. This method is simpler and more compatible with other styles, and will also produce a list that is correctly sorted by the shorthand. (Cf. niso:bibref, bsi:abbreviation, iso:electrodoc.)

I should clarify here that this automatic placement of the shorthand at the head of the entry will *not* occur if you set the package option cmslos=false in your preamble. This allows you to implement other systems of shorthand expansion using either a list of shorthands (via \printshorthands, which is always available no matter what the state of cmslos) or cross-references (via customc) within the reference list itself. You can place skiplos in the options field to exclude a particular entry from the list of shorthands if you do decide to print that list, giving maximum flexibility.

Indeed, I have provided two options to add to this flexibility. First, I have included two bibenvironments for use with the env option to the \printshorthands command: losnotes is designed to allow a list of shorthands to appear inside footnotes, while losendnotes does the same for endnotes. Their main effect is to change the font size, and in the

latter case to clear up some spurious punctuation and white space that I see on my system when using endnotes. (You'll probably also want to use the option heading=none in order to get rid of the [oversized] default, providing your own within the \footnote command.) Second, I have provided a package option, shorthandfull, which prints entries in the list of shorthands which contain full bibliographical information, effectively allowing you to eschew the list of references in favor of a fortified shorthand list. This option will only work if used in tandem with cmslos=false, as otherwise the shorthand will be printed twice. (See 15.37, 13.67, 14.59–60, and also biblatex.pdf for more information.)

As I mentioned above under **crossref**, I believe it is safe to use shorthands in parent entries, as this, in the standard configuration, gives you the shorthand itself in the child entry's abbreviated cross-reference, which may well save space in the list of references.

shortjournal A special biblatex field, used to provide both an abbreviated form of a journaltitle in citations and/or the reference list and to facilitate the creation of a list of journal abbreviations, should this be needed, rather in the manner of a shorthand list. As requested by user BenVB, you can now utilize this functionality in your documents, but there are a few details worth mentioning here. First, users in some fields may well already be accustomed to using a set of standard journal abbreviations (15.46), in which case the journaltitle field may well already contain the abbreviation, which will appear wherever that field is printed. In such cases, it usually isn't necessary to provide a list of abbreviations in individual publications, but were you to require such a thing, you'd have to move the abbreviation from the journaltitle to the shortjournal field, placing the full title in the former. In periodical entries the title field presents what would be the journaltitle in the articles or reviews, so in such entries you can provide the standard shorttitle field to accompany the title, and biblatex-chicago will automatically copy the shorttitle into a shortjournal.

Having done this, you then need to choose where to print the shortjournal, which is controlled by the journalabbrev option either in the preamble or in the options field of individual .bib entries. By default, and taking account of the space-saving features of the author-date styles, this option is set to notes, so your shortjournal fields will be printed only in those citations where they appear in place of an author. There are three other settings: true prints the shortened fields both in citations and in the reference list, bib prints them only in the reference list, and false ignores them. Should you wish to present a list of these abbreviations with their expansions, then you need to use the \printbiblist{shortjournal} command, perhaps with a title option to differentiate the list from any shorthand list. As with shorthand lists, I have provided two bibenvironments for printing this list in foot- or endnotes (sjnotes and sjendnotes, respectively), to be used with the env option to \printbiblist. Again as with shorthands, you'll probably want to use the option heading=none when using these environments, just to turn off the (oversized) default, and perhaps provide your own title within the \footnote command. Finally, if you don't like the default formatting of the abbreviations in the list (bold italic), you can change it with \DeclareFieldFormat{shortjournal-width} — you can see its default definition at the top of chicago-authordate.bbx.

shortseries A special biblatex field, used both to provide an abbreviated form of a (book) series in a reference list and to facilitate the creation of a list of such abbreviations rather in the manner of a shorthand list. As with the shortjournal field, its inclusion in biblatex-chicago was requested by user BenVB, and it is now available in entry types which have book-like series titles rather than journal-like numbers in the series field, to wit: audio, book, bookinbook, collection, inbook, incollection, inproceedings, inreference, letter, manual, music, mvbook, mvcollection, mvproceedings, mvreference, reference, report, standard, suppbook, and video. There are several steps to take in order to use the field. First, you'll need to provide both shortseries and series fields in the entry, then you'll need to set the seriesabbrev option either when loading biblatex-chicago, for the whole document or for specific entry types, or in the options field of individual .bib entries. By default, this option is not set, so your shortseries fields will be silently ignored. Setting it to true prints the shortened fields in the reference list. Should you wish to present a list of these abbreviations with their expansions, then you need to use the \printbiblist{shortseries} command, perhaps with a title option to differentiate the list from any shorthand list. As

with shorthand lists, I have provided two bibenvironments for printing this list in foot- or endnotes (shsernotes and shserendnotes, respectively), to be used with the env option to \printbiblist. Again as with shorthands, you'll probably want to use the option heading=none when using these environments, just to turn off the (oversized) default, and perhaps provide your own title within the \footnote command. Finally, if you don't like the default formatting of the abbreviations in the list (plain roman), you can roll your own using \DeclareFieldFormat{shortserieswidth} — you can see its default definition at the top of chicago-authordate.bbx.

shorttitle A standard biblatex field, primarily used to provide an abbreviated title for citation styles that need one. (It is also the way to hook periodical entries into the shortjournal mechanism, on which see the previous entry.) In biblatex-chicago-authordate such a field will be necessary only very rarely (unlike in the notes & bibliography style), and is most likely to turn up in inreference or reference entries (where the title takes the place of the author), in dataset entries, or in any sort of entry with a classical entrysubtype or with authortitle set in its options field. These latter three contexts make citations use author and title instead of author and year, and if an abbreviated version of that title would save space in your running text this is the field where you can provide it. (Cf. ency:britannica, grove:sibelius, aristotle:metaphy:gr.)

sortkey
sortname
sorttitle
sortyear Standard biblatex fields, designed to allow you to specify how you want an entry alphabetized in a list of references. The sortkey field trumps all other sorting information, while the others offer more fine-grained control. In general, if an entry doesn't turn up where you expect or want it, one of these fields should provide the solution. Entries with a corporate author can omit the definite or indefinite article, which should help (14.70, 14.84; cotton:manufacture, nytrumpet:art). The default settings of \DeclareSortingTemplate include the three supplemental name fields (name[a-c]) and also the journaltitle in the sorting algorithm, so once again you should find those algorithms needing less help than before. Entries headed by a title beginning with the definite or indefinite article may well still require such assistance (grove:sibelius). There may be circumstances — several reprinted books by the same author, for example — when the sortyear field is the best choice. Please consult biblatex.pdf for the details.

subtitle The subtitle for a title — see next entry.

timezone This field can, if necessary, specify the time zone associated with a time stamp given as part of an date. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

title Biblatex-chicago includes the authordate-trad style, designed as a kind of hybrid style according to indications contained in the *Manual* (15.38). This trad style differs *only* in the way it treats the title and related fields, which retain the forms they have traditionally had in the Chicago author-date specifications prior to the 16th edition. Where newer editions use headline-style capitalization, the older editions used sentence-style; where newer editions place article or incollection titles within quotation marks, the older editions presented them in plain text. I include below, under a separate rubric, full documentation of trad title fields for those needing or wishing to use them. First, though, I document the same field(s) for the standard author-date style.

In the vast majority of cases, this field works just as it always has in BIBTEX, and just as it does in biblatex. The *Manual* recommends that titles be treated more or less identically across both its systems of documentation (15.3, 15.6, 15.13). This means that users of the author-date style don't need to worry about sentence-style capitalization when compiling their .bib databases, and so can eschew the extra curly braces needed to preserve uppercase letters in this context. These rules, however, mean that a few complications familiar to users of the notes & bibliography style do arise. First, although nearly every entry will have a title, there are some exceptions, particularly incollection or online entries with a merely generic title, instead of a specific one (centinel:letters, powell:email). Second, the *Manual*'s rules for formatting titles, which also hold for booktitles and maintitles, require additional attention. The whole point of using a biblatex-based system is

for it to do the formatting for you, and in most cases biblatex-chicago-authordate does just that, surrounding titles with quotation marks, italicizing them, or occasionally just leaving them alone. When, however, a title is quoted within a title, then you need to know some of the rules. A summary here should serve to clarify them, and help you to understand when biblatex-chicago-authordate might need your help in order to comply with them.

The internal rules of biblatex-chicago-authordate are as follows:

Italics: booktitle, maintitle, and journaltitle in all entry types; title of artwork, book, bookinbook, booklet, collection, image, manual, misc (with no entrysubtype), performance, periodical, proceedings, report, standard, suppbook, and suppcollection entry types.

Quotation Marks: title of article, inbook incollection, inproceedings, online, periodical, thesis, and unpublished entry types, issuetitle in article, periodical, and review entry types.

Unformatted: booktitleaddon, maintitleaddon, and titleaddon in all entry types, title of customc, letter, misc (with an entrysubtype), patent, review, and suppperiodical entry types.

Italics or Quotation Marks: All of the audiovisual entry types — audio, music, and video — have to serve as analogues both to book and to inbook. Therefore, if there is both a title and a booktitle, then the title will be in quotation marks. If there is no booktitle, then the title will be italicized, unless you provide an entrysubtype.

Now, the rules for which entry type to use for which sort of work tend to be fairly straightforward, but in cases of doubt you can consult section 5.1 above, the examples in datestest.bib, or go to the *Manual* itself, 8.156–201. Assuming, then, that you want to present a title within a title, and you know what sort of formatting each of the two would, on its own, require, then the following rules apply:

1. Inside an italicized title, all other titles are enclosed in quotation marks and italicized, so in such cases all you need to do is provide the quotation marks using \mkbibquote, which will take care of any following punctuation that needs to be brought within the closing quotation mark(s) (14.94; donne:var, mchugh:wake).

2. Inside a quoted title, you should present another title as it would appear if it were on its own, so in such cases you'll need to do the formatting yourself. Within the double quotes of the title another quoted title would take single quotes — the \mkbibquote command does this for you automatically, and also, I repeat, takes care of any following punctuation that needs to be brought within the closing quotation mark(s). (See 14.94–95; garrett, loften:hamlet, murphy:silent, white:callimachus.)

3. Inside a plain title (most likely in a review entry or a titleaddon field), you should present another title as it would appear on its own, once again formatting it yourself using \mkbibemph or \mkbibquote. (barcott:review, gibbard, osborne:poison, ratliff:review, unsigned:ranke).

The *Manual* provides a few more rules, as well. A word normally italicized in text should also be italicized in a quoted or plain-text title, but should be in roman ("reverse italics") in an italicized title. A quotation used as a (whole) title (with or without a subtitle) retains, according to the 16th edition, its quotation marks in an italicized title if it appears that way in the source, but I can't find similar instructions in the 17th. Such a quotation always retains its quotation marks when the surrounding title is quoted or plain (14.94; lewis). A word or phrase in quotation marks, but that isn't a quotation, retains those marks in all title types (kimluu:diethyl).

Finally, please note that in all review (and suppperiodical) entries, and in misc entries with an entrysubtype, and only in those entries, biblatex-chicago-authordate will automatically capitalize the first word of the title after sentence-ending punctuation, assuming

that such a title begins with a lowercase letter in your .bib database. See **\autocap** in section below for more details.

title (trad) When you choose the authordate-trad style, your title and related fields will need extra care, familiar to users of the 15th-edition author-date style. The whole point of using a biblatex-based system is for it to do the formatting for you, and in most cases biblatex-chicago-authordate-trad does just that, capitalizing titles sentence-style, italicizing them, and sometimes both. There are two situations that require user intervention. First, in titles that take sentence-style capitalization, you need, as always in traditional BIBTEX, to assist the algorithms by placing anything that needs to remain capitalized within an extra pair of curly braces. Second, when a title is quoted within a title, you need to know some of the rules of the Chicago style. A summary here should serve to clarify them, and help you to understand when biblatex-chicago-authordate-trad might need your help in order to comply with them.

With regard to sentence-style capitalization, the rules of the Chicago authordate-trad style are fairly simple:

> **Headline Style:** journaltitle in all types, series in all book-like entries (i.e., not in articles), and title in periodical entries.
>
> **Sentence Style:** every other title, *except* in letter entries, review and suppperiodical entries, and in misc entries with an entrysubtype. Also, the booktitle, issuetitle, and maintitle in all entry types use sentence style.
>
> **Contextual Capitalization of First Word:** titleaddon, booktitleaddon, maintitleaddon in all entry types, also the title of review entries, of suppperiodical entries, and of misc entries with an entrysubtype.
>
> **Plain:** title in letter entries.

What this means in practice is that to get a title like *The Chicago manual of style*, your .bib entry needs to have a field that looks something like this:

```
title = {The {Chicago} Manual of Style}
```

This is completely straightforward, but remember that if an article has a title like: Review of *The Chicago manual of style*, then the curly braces enclosing material to be formatted in italics will cause the capitalization algorithm to stop and leave all of that material as it is, so your .bib entry would need to have a field something like this:

```
title = {\bibstring{reviewof} \mkbibemph{The Chicago manual of style}}
```

(As an aside, the use of the reviewof bibstring isn't strictly necessary here, but it helps with portability across languages and across the two Chicago styles. If you've noticed a lot of lowercase letters starting fields in dates-test.bib, they're present because in the notes & bibliography style capitalization is complicated by notes using commas where the bibliography uses periods, and words like "review" start in uppercase only if the context demands it. There's considerably less of this in the author-date styles [note the *titleaddon fields], but it still pays to be aware of the issue.)

With regard to italics, the rules of biblatex-chicago-authordate-trad are as follows:

> **Italics:** booktitle, maintitle, and journaltitle in all entry types; title of artwork, book, bookinbook, booklet, collection, manual, misc (w/o entrysubtype), performance, periodical, proceedings, report, standard, suppbook, and suppcollection types.
>
> **Main Text Font (Roman):** title of article, image, inbook, incollection, inproceedings, letter, misc (with an entrysubtype), online, patent, periodical, review, suppperiodical, thesis, and unpublished entry types, issuetitle in article and periodical entry types. booktitleaddon, maintitleaddon, and titleaddon in all entry types.
>
> **Italics or Roman:** All of the audiovisual entry types — audio, music, and video — have to serve as analogues both to book and to inbook. Therefore, if there is both a title and a booktitle, then the title will be in the main text font. If there is no booktitle, then the title will be italicized, unless you provide an entrysubtype.

Now, the rules for which entry type to use for which sort of work tend to be fairly straightforward, but in cases of doubt you can consult section 5.1 above, the examples in dates-test.bib, or go to the *Manual* itself, 8.156–201. Assuming, then, that you want to present a title within a title, and you know what sort of formatting each of the two would, on its own, require, then the following rules apply:

1. Inside an italicized title, all other titles are enclosed in quotation marks and italicized, so in such cases all you need to do is provide the quotation marks using `\mkbibquote`, which will take care of any following punctuation that needs to be brought within the closing quotation mark(s) (14.94; donne:var, mchugh:wake).

2. Inside a plain-text title, you should set off other plain-text titles with quotation marks, while italicized titles should appear as they would if they were on their own. In such cases you'll need to do the formatting yourself, using `\mkbibemph` or `\mkbibquote`. (See barcott:review, garrett, gibbard, loften:hamlet, loomis:structure, murphy:silent, osborne:poison, ratliff:review, unsigned:ranke, white:callimachus.)

The *Manual* provides a few more rules, as well. A word normally italicized in text should also be italicized in a plain-text title, but should be in roman ("reverse italics") in an italicized title. A quotation used as a (whole) title (with or without a subtitle) retains, according to the 16th edition, its quotation marks in an italicized title if it appears that way in the source, but I can't find similar instructions in the 17th. Such a quotation always retains its quotation marks when the surrounding title is quoted or plain (14.94; lewis). A word or phrase in quotation marks, but that isn't a quotation, retains those marks in all title types (kimluu:diethyl).

Finally, please note that there is also a preamble option — `headline` — that disables the automatic sentence-style capitalization routines in authordate-trad. If you set this option, the word case in your title fields will not be changed in any way, that is, this doesn't automatically transform your titles into headline-style, but rather allows the .bib file to determine capitalization. It works by redefining the command `\MakeSentenceCase`, so in the unlikely event you are using the latter anywhere in your document please be aware that it will also be turned off there. See section 5.4.3, below.

titleaddon Standard biblatex intends this field for use with additions to titles that may need to be formatted differently from the titles themselves, and biblatex-chicago uses it in just this way, with the additional wrinkle that it can, if needed, replace the title entirely, and this in, effectively, any entry type, providing a fairly powerful, if somewhat complicated, tool for getting biblatex to do what you want (cf. centinel:letters). This field will always be unformatted, that is, neither italicized nor placed within quotation marks, so any formatting you may need within it you'll need to provide manually yourself. The single exception to this rule is when your data begins with a word that would ordinarily only be capitalized at the beginning of a sentence, in which case you need then simply ensure that that word is in lowercase, and biblatex-chicago will automatically do the right thing. See **\autocap** in section 5.3.1 below. The package and entry options ptitleaddon and ctitleaddon (section 5.4.2) can help you customize the punctuation that appears before the titleaddon field. Please note, however, that I have added this field to the periodical entry type, and that the punctuation there is governed by the jtitleaddon option, which defaults to a space. (Cf. brown:bremer, osborne:poison, reaves:rosen, and white:ross:memo for examples where the field starts with a lowercase letter; morgenson:market provides an example where the titleaddon field, holding the name of a regular column in a newspaper, is capitalized, a situation that is handled as you would expect; coolidge:speech shows an entry option for controlling the punctuation.)

translator As far as possible, I have implemented this field as biblatex's standard styles do, but the requirements specified by the *Manual* present certain complications that need explaining. Biblatex.pdf points out that the translator field will be associated with a title, a booktitle, or a maintitle, depending on the sort of entry. More specifically, biblatex-chicago associates the translator with the most comprehensive of those titles, that is, maintitle if there is one, otherwise booktitle, otherwise title, if the other two are lacking. In a large number of

cases, this is exactly the correct behavior (adorno:benj, centinel:letters, plato:republic:gr, among others). Predictably, however, there are numerous cases that require, for example, an additional translator for one part of a collection or for one volume of a multi-volume work. For these cases I have provided the nameb field. You should format names for this field as you would for author or editor, and these names will always be associated with the title (euripides:orestes). In the algorithm for finding a name for the head of a reference list entry or for a citation, nameb takes precedence over translator.

I have also provided a namea field, which holds the editor of a given title (euripides:orestes). If namea and nameb are the same, biblatex-chicago will concatenate them, just as biblatex already does for editor, translator, and namec (i.e., the compiler). Furthermore, it is conceivable that a given entry will need separate translators for each of the three sorts of title. For this, and for various other tricky situations, there is the \parttrans macro (and its siblings), designed to be used in a note field or in one of the titleaddon fields (ratliff:review). (Because the strings identifying a translator differ in notes and bibliography, one can't simply write them out in such a field when using the notes & bibliography style, but you can certainly do so in the author-date styles, if you wish. Using the macros will make your .bib file more portable across both Chicago specifications, and also across multiple languages, but they are otherwise unnecessary. [See section 7].)

Finally, as I detailed above under **author**, in the absence of an author, a namea, an editor, and a nameb, the translator will be used at the head of an entry (silver:gawain), and the reference list entry alphabetized by the translator's name, behavior that can be controlled with the use<name> switches in the options field. Cf. author, editor, namea, nameb, and namec.

type This is a standard biblatex field, and in its normal usage serves to identify the type of a manual, patent, report, or thesis entry. Biblatex implements the possibility, in some circumstances, to use a bibstring without inserting it in a \bibstring command, and in some entry types (audio, manual, music, patent, report, suppbook, suppcollection, thesis, and video) the type field works this way, allowing you simply to input, e.g., patentus rather than \bibstring{patentus}, though both will work. (See petroff:impurity; herwign:office, murphy:silent, and ross:thesis all demonstrate how the type field may sometimes be automatically set in such entries by using one of the standard entry-type aliases). In other entry types (artwork, image, book, online, article, review, and suppperiodical) biblatex-chicago will merely capitalize the contents according to context.

Another use for the field is to generalize the functioning of the suppbook entry type, and of its alias suppcollection. In such entries, the type field can specify what sort of supplemental material you are citing, e.g., "preface to" or "postscript to." Cf. suppbook above for the details. (See *Manual* 14.110; polakow:afterw, prose:intro).

You can use the type field in artwork, audio, image, music, and video entries to identify the medium of the work, e.g., oil on canvas, albumen print, compact disc or MPEG. In book entries it will normally hold system information about multimedia app content, while in online, article, and review entries it will hold the medium of online multimedia (15.57, 14.267–68). Cf. under these entry types in section 5.1, above, for more details. (See auden:reading, bedford:photo, cleese:holygrail, leo:madonna, nytrumpet:art.)

url Standard biblatex field, it holds the url of an online publication, though you can provide one for all entry types. The *Manual* expresses a strong preference for DOIs over URLs if the former is available — cf. doi above, and also urldate just below. The required LaTeX package url will ensure that your documents format such references properly, in the text and in the reference apparatus. It may be worth noting that child entries no longer inherit url fields from their parents — the information seems entry-specific enough to warrant a little bit of extra typing if you need to present the same locator in several entries.

urldate Standard biblatex field, it identifies exactly when you accessed a given url. The *Manual* prefers DOIs to URLs; in the latter case it allows the use of access dates, particularly in contexts that require it, but prefers that you use revision dates, if these are available. To enable you to specify which date is at stake, I have provided the **userd** field, documented

below. If an entry doesn't have a userd, then the urldate will be treated as an access date (14.8, 14.12–13, 15.50; evanston:library, grove:sibelius, hlatky:hrt, osborne:poison, sirosh:visualcortex, wikiped:bibtex). In the default setting of \DeclareLabeldate, any entry without a date, eventdate, or origdate will use the urldate to find a year for citations and the list of references (grove:sibelius, wikiped:bibtex), but *only* if the urldate isn't an access date, that is, only if a userd field is present. If the only date available is an online access date, then the entry is considered to have no date, and "n.d." will appear instead, though of course the access date will still be printed later in the reference list entry. (If you were to put the string accessed into the userd field, you could work around this prohibition.)

You can also use the urldate field to specify a time stamp, should the date alone not be specific enough. The time stamp follows the date, separated by an uppercase "T", like so: yyyy-mm-ddThh:mm:ss. If you wish to specify the time zone, the *Manual* (10.41) prefers initialisms like "EST" or "PDT," and these are most easily provided using the urltimezone field, where you can provide your own parentheses if so desired (cp. 14.191). Following the examples in the *Manual*, any urldate will by default be printed in 24-hour format, though other time stamps use 12-hour format. The biblatex option urltime, discussed in section 5.4.1, allows you to change this in your preamble.

urlstamp
A urldate time stamp (and urltimezone) can appear in any entry whatsoever, if you judge the online source to be the sort that changes rapidly enough for a time stamp to be necessary (14.207, 14.233; wikiped:bibtex). You can stop it printing by setting the new urlstamp option to false in your preamble for the whole document or for specified entry types, or in the options field of individual entries. Please see the documentation of **date** and also table 3, above, for more details about time stamps and other parts of biblatex's enhanced date specifications. Table 2 contains a summary of the current state of biblatex-chicago's handling of online materials.

urltimezone
This field can, if necessary, specify the time zone associated with a time stamp given as part of an urldate. The *Manual* prefers initialisms like "EST" for this purpose, and you can provide parentheses around it at your discretion (cp. 10.41 and 14.191).

usera
A supplemental biblatex field which in certain contexts in biblatex-chicago will identify the broadcast network when you cite a radio or television program. In article, periodical, and review entries with entrysubtype magazine, it acts almost as a "journaltitleaddon" field, and its contents will be placed, unformatted and between commas, after the journaltitle and before the date. In video entries it comes after the eventdate, i.e., the date of first broadcast, and is separated from that date by the \bibstring "on" (14.213, 14.265; american:crime, bundy:macneil, friends:leia, mayberry:brady).

userc
I have implemented this supplemental biblatex field as part of the Chicago author-date style's handling of cross-references within the list of references. (The "c" part is meant as a sort of mnemonic for this latter function.) In recent editions of the *Manual* you no longer need to use the **customc** entry type to include alphabetized expansions of shorthands in the reference list, but you may still need to provide cross-references of some sort to separate entries in that list, perhaps when a single author uses multiple pseudonyms. In such a case it is unlikely that you will cite the customc entry itself in the body of your text. Therefore, in order for it to appear in the reference list, you have two choices. You can either include the entry key of the customc entry in a \nocite command inside your document, or you can place that entry key in the userc field of the .bib entry that actually contains one of the full citations. In the latter case, biblatex-chicago will call \nocite for you when you cite the main entry. (See 14.81–82; creasey:ashe:blast, creasey:morton:hide, creasey:york:death, lecarre:quest.)

userd
The userd field acts as a sort of "datetype" field, allowing you in most entry types to identify whether a urldate is an access date or a revision date. The general usage is fairly simple. If this field is absent, then a urldate will be treated as an access date, as has long been the default in biblatex and in biblatex-chicago. If you need to identify it in any other way, what you include in userd will be printed *before* the urldate, so phrases

like "last modified" or "last revised" are what the field will typically contain (14.12–13; wikiped:bibtex). In the absence of a urldate, you can in most entry types include a userd field to qualify a date in the same way it would have modified a urldate. If an entry contains *only* a urldate and no other sort of date, and has no userd field, that entry will now be treated as though it had no date, and "n.d." will appear in citations and at the head of entries in the reference list (15.50).

Because of the rather specialized needs of some audio-visual references, this basic schema changes for music and video entries. In music entries where an eventdate is present, userd will modify that date instead of any urldate that may also be present, and it will modify an origdate if it is present and there is no eventdate. It will modify a date only in the absence of the other three. In video entries it will modify an eventdate if it is present, and in its absence the urldate. Given the absence of those two, it can modify a date. In all these cases, userd will modify what remains of any date, i.e., the month and the day, if that date's year has been printed at the head of the entry. Please see the documentation of the music and video entry types, and especially of the eventdate, origdate, and urldate fields, above (14.276–279, 15.53; nytrumpet:art).

In all cases, you can start the userd field with a lowercase letter, and biblatex will take care of automatic contextual capitalization for you.

usere Another supplemental biblatex field, which biblatex-chicago uses specifically to provide a translated title of a work, something that may be needed if you deem the original language unparseable by a significant portion of your likely readership. The *Manual* offers two alternatives in such a situation: either you can translate the title and use that translation in your title field, providing the original language in language, or you can give the original title in title and the translation in usere. Cf. **language**, above. (See 14.99; kern, pirumova:russian, weresz.)

userf See section , below.

venue Standard biblatex offers this field for use in proceedings and inproceedings entries, and after a request from Patrick Danilevici I have followed suit. I have also implemented the field in the misc entry type, both with and without an entrysubtype, in the performance type, and in the unpublished type. In all uses it will normally present the actual venue of an event, as opposed, e.g., to the origlocation, which might present where a letter was written or where an earlier edition was printed.

verbc Author-date styles in biblatex use the extradate field, automatically provided by biber, to distinguish citations of different works by the same author that were published in the same year, e.g., (Surname 1978**a**). The Chicago author-date styles recommend that some sorts of material — online comments, newspaper articles, and live performances, *inter alia* — needn't appear in reference lists, but only in the text, often accompanied by a full date reference (cmsdate=full) rather than by the rather less informative year on its own. In most circumstances a simple skipbib in the options field will suffice, but, especially with online materials, it is possible, even probable, that users will have .bib databases containing different works by the same author from the same year, only *some* of which need to appear in the reference list. Biber will provide extradate fields for all these entries, however, so it is easy to get an extradate letter in a reference even when only one work by that author appears in the list, or perhaps a series of letters with some missing from the sequence.

The verbc field allows you manually to intervene to control these side effects. (It's a standard biblatex field, but isn't used in the standard styles.) Putting anything in the field prevents that entry from interfering in the extradate provision of entries that don't contain such a field, and in more complicated scenarios you could group entries by identical verbc field to prevent them from interfering both with entries not having any verbc field *and* with entries having a different value for that field. By default, the commenton relatedtype for online and review entries adds a verbc field to its entry, but you can in all cases control this and provide your own in any circumstances and in any entry type you

wish. Please see the documentation of those two entry types in section 5.1, and of the commenton relatedtype in section 5.2.1.

version   Standard biblatex field, formerly only available in artwork, image, misc, music, and patent entries in biblatex-chicago-authordate, but now also in book and performance entries. In most entry types it prints a localized "version" string, but there may be specialist needs in artwork and image entries, so there you'll need to specify the type of data inside the field itself. In the book type it is particularly needed for presenting multimedia app content (15.57, 14.268).

volume   Standard biblatex field. It holds the volume of a journaltitle in article entries, and also the volume of a multi-volume work in many other sorts of entry. The treatment and placement of volume information in book-like entries is rather complicated in the *Manual* (14.116–22, 15.41). In the reference list, the volume appears either before the maintitle or before the publication information, while in citations you may need to provide it in the postnote field — see the volumes field, just below. In a number of these contexts, and in both books and periodicals, volume information can appear *immediately before* the page number(s). In such a case, the *Manual* (14.116) prescribes the same treatment for both sorts of sources, that is, that "a colon separates the volume number from the page number with no intervening space." I have implemented this, but at the request of Clea F.

\postvolpunct   Rees I have made this punctuation customizable, using the command \postvolpunct. By default it prints \addcolon, so use \renewcommand{\postvolpunct}{…} in your preamble to redefine it. Cf. part, and the command documentation in section 5.3.1; conway:evolution shows how sometimes this field may hold series information, as well.

volumes   Standard biblatex field. It holds the total number of volumes of a multi-volume work, and in such references you should provide the volume and page numbers in the postnote field of the relevant \cite command, e.g.:

\autocite[3:25]{bibfile:key}.

Cf. 15.22; meredith:letters, tillich:system, weber:saugetiere, wright:evolution. The entry wright:theory presents one volume of such a multi-volume work, so you would no longer need to give the volume in any postnote field when citing it. If both a volume and a volumes field are present, as may occur particularly in cross-referenced entries, then biblatex-chicago will ordinarily suppress the volumes field, except in some cases when a maintitle is present. In this latter case, if the volume appears before the maintitle, the

hidevolumes   option hidevolumes, set to true by default, controls whether to print the volumes field after that title or not. Set it to false either in the preamble or in the options field of your entry to have it appear after the maintitle. See the option's documentation in section 5.4.2, below.

xref   A modified crossref field provided by biblatex, which prevents inheritance of any data from the parent entry. See **crossref**, above.

year   Standard biblatex field, especially important for the author-date specification. Please see all the details under **date** above. Unlike the date field year allows non-numeric input, so you can put \bibstring{nodate} here if required, or indeed any other sort of non-numerical date information. For many kinds of uncertain and unspecified dates it is now much simpler to make use of biblatex's enhanced date specifications in the date field, instead. Please see table 3 for a summary of how biblatex-chicago implements these enhancements. Cf. bedford:photo, clark:mesopot, leo:madonna, ross:thesis.

5.2.1   Fields for Related Entries

As biblatex.pdf puts it (§ 3.4), "Almost all bibliography styles require authors to specify certain types of relationship between entries such as 'Reprint of,' 'Reprinted in,' etc. It is impossible to provide data fields to cover all of these relationships and so biblatex provides a general mechanism for this using the entry fields related, relatedtype and relatedstring." Before this mechanism was available biblatex-chicago attempted to provide a

similar but much more limited set of inter-entry relationships using the biblatex fields origlanguage, origlocation, origpublisher, pubstate, reprinttitle, and userf. All of these still work just as they always have or, I hope, somewhat better than they always have after many recent bug fixes, but the more general and more powerful biblatex related mechanism is also available. It can provide much of what the older system provided and a great deal that it couldn't. What follows is a field-by-field discussion of the options now available.

origlanguage In keeping with the *Manual*'s specifications, I have fairly thoroughly redefined biblatex's facilities for treating translations. The origtitle field isn't used, while the language and origdate fields have been press-ganged for other duties. The origlanguage field, for its part, retains a dual role in presenting translations in a list of references. The details of the *Manual*'s suggested treatment when both a translation and an original are cited may be found below under **userf**. Here, however, I simply note that the introductory string used to connect the translation's citation with the original's is "Originally published as," which I suggest may well be inaccurate in a great many cases, as for instance when citing a work from classical antiquity, which will most certainly not "originally" have been published in the Loeb Classical Library. Although not, strictly speaking, authorized by the *Manual*, I have provided another way to introduce the original text, using the origlanguage field, which must be provided *in the entry for the translation, not the original text* (aristotle:metaphy:trans). If you put one of the standard biblatex bibstrings there (enumerated below), then the entry will work properly across multiple languages. Otherwise, just put the name of the language there, localized as necessary, and biblatex-chicago will eschew "Originally published as" in favor of, e.g., "Greek edition:" or "French edition:". This has no effect in citations, where only the work cited — original or translation — will be printed, but it may help to make the *Manual*'s suggestions for the list of references more palatable. **NB:** You can use the relatedtype origpubas with a customized relatedstring field to achieve the same ends.

That was the first usage, in keeping at least with the spirit of the *Manual*. I have also, perhaps less in keeping with that specification, retained some of biblatex's functionality for this field. If an entry doesn't have a userf field, and therefore won't be combining a text and its translation in the list of references, you can also use origlanguage as biblatex intended it, so that instead of saying, e.g., "translated by X," the entry will read "translated from the German by X." The *Manual* doesn't mention this, but it may conceivably help avoid certain ambiguities in some citations. As in biblatex, if you wish to use this functionality, you have to provide *not* the name of the language, but rather a bibstring, which may, at the time of writing, be one of american, brazilian, danish, dutch, english, french, german, greek, italian, latin, norwegian, portuguese, spanish, or swedish, to which I've added russian.

origlocation This field mainly serves to help document reprint editions and their corresponding originals (14.114, 15.40). In biblatex-chicago you can provide both an origlocation and an origpublisher to go along with the origdate, should you so wish, and all of this information will be printed in the reference list. You can also use this field in a letter or misc (with entrysubtype) entry to give the place where a published or unpublished letter was written (14.111, 14.229). (Jonathan Robinson has suggested that the origlocation may in some circumstances actually be helpful for disambiguation, his example being early printed editions of the same material printed in the same year but in different cities. The new functionality should make this simple to achieve. Cf. origdate [section 5.2], origpublisher and pubstate; schweitzer:bach.) **NB:** It is impossible to present this same information, as here, *inside* a single entry using a related field, though the relatedtype origpubin presents much the same information *after* the entry, using data extracted from a separate entry.

origpublisher As with the origlocation field just above, this field mainly serves to help document reprint editions and their corresponding originals (14.114, 15.40). You can provide an origpublisher and/or an origlocation in addition to the origdate, and all will be presented in the reference list. (Cf. origdate [section 5.2], origlocation, and pubstate; schweitzer:bach.) **NB:** It is impossible to present this same information, as here, *inside* a single entry using a re-

lated field, though the relatedtype origpubin presents much the same information *after* the entry, using data extracted from a separate entry.

pubstate In response to new specifications in the 17th edition of the *Manual* (esp. 14.137), I have tried to generalize the functioning of the pubstate field in all entry types. Because the author-date style has fairly complicated rules about presenting reprinted editions (15.40), the reprint string still has a special status. Depending on which date(s) you have chosen to appear at the head of the entry, biblatex-chicago-authordate will either print the (localized) string reprint in the proper place or otherwise provide a notice at the end of the entry detailing the original publication date. See under **date** above for the available permutations. (Cf. aristotle:metaphy:gr, maitland:canon, maitland:equity, schweitzer:bach.)

NB Other strings are divided into two types: those which biblatex-chicago will print as the year, which currently means *only* those for which biblatex contains bibstrings indicating works soon to be published, i.e., forthcoming, inpreparation, inpress, and submitted; and those, i.e., everything else, which will be printed before, and in close association with, other information about the publisher of a work. (This is a change from previous behavior, where non-reprint strings were printed *after* the publication information, as in the standard styles. You can still use the addendum field to present information here, of course.) The four strings that replace the year will always be localized, as will reprint and selfpublished (and anything else that biblatex finds to be a \bibstring) from the second category. All other strings will be printed as-is, capitalized if needed, just before the publisher (author:forthcoming, contrib:contrib, schweitzer:bach).

There is one further subtlety of which you ought to be aware. In music and video entries, the reprint string in pubstate will only make a difference to your entries when the date which it modifies — the origdate, typically — *doesn't* appear in citations and at the head of reference-list entries. In this case the date is treated as an original release date, and it will be printed, preceded by the appropriate string, near the end of the entry. Other strings don't show this special behavior in these entries. **NB:** For those uses of the pubstate field that print a notice at the end of the entry, the relatedtype origpubin provides much the same information, using data extracted from a different entry. If the information appears inside the entry then there is no equivalent related functionality.

related This field is required to use biblatex's related functionality, and it should contain the entry key or keys from which biblatex should extract data for presentation not on its own, but rather in the reference list entry which contains the related field itself. Indeed, unless you change the defaults using the relatedoptions field this data will only appear in such entries, never on its own and never in citations. Without a relatedtype field, this will print the default type, equivalent to a full reference list entry *immediately after* the entry containing the related field, with no intervening string. You can specify a string using the relatedstring field, so in effect this presents a powerful mechanism for presenting full references to related material of any sort whatsoever.

related=true By default, the package option related is set to print related entries in the list of references. If you would like to turn this off you can set this option, either in your preamble or in the options or relatedoptions field of the relevant entry, to false. For the two relatedtypes that construct a single entry using data extracted from related entries — commenton and reviewof — you'll need to make sure this is true to get properly-formatted citations in the reference list. See below for the details.

relatedoptions This field will, I should expect, only be needed very rarely. If you want to set entry-level options for a related entry this is where you can do it, though please remember one important detail. By default, Biber sets this option to dataonly, which among other things prevents the related entry from appearing separately in the list of references, assuming you don't specifically cite it elsewhere. If you use the field yourself, then you'll need to include dataonly as one of the options therein to maintain this effect. Of course, it may be you don't want all the effects of dataonly, so you can tailor it however you wish. See biblatex.pdf § 3.4.

relatedstring The procedure for choosing a string to connect the main entry with its related entry/ies is straightforward, the default being a bibstring, if any, with the same name as the relatedtype, or alternately a string or strings defined within the driver for that relatedtype, as

happens with the types origpubin and bytranslator. Failing these, you can supply your own in the relatedstring field, either in the form of the name of a pre-defined bibstring or as any text you choose, and anything in this field always takes precedence over the automatic choices. If your non-bibstring starts with a lowercase letter then biblatex-chicago will capitalize it automatically for you depending on context (coolidge:speech, weed:flatiron). I have not altered the standard relatedtype strings, and have in fact modified the reprinttitle mechanism to use the reprintfrom string, which works better syntactically in this context, and modified the pubstate mechanism to use the origpubin string, which brings it into line with the notes & bibliography style.

relatedtype   The standard biblatex styles define six relatedtypes, and I have either simply adopted them wholesale or adapted them to the needs of the Chicago style, retaining the basic syntax as much as possible. I have also added three to these six (see below):

> **bytranslator:** This prints a full reference to a translation, starting with the (localized) string "Translated by translator as Title, ..." The reference is fuller in biblatex-chicago than in the standard styles, and for the first time allows users to choose the *Manual's* alternate method for presenting original + translation (14.99; furet: passing:fr). The old userf mechanism provides the other, as does the origpubas relatedtype (see below).

> **default:** This is the macro used when no relatedtype is defined. It prints, as in the standard styles, and with no intervening string, full references to the related entries.

> **multivolume:** This briefly lists the individual volumes in a multi-volume work, and works much as in the standard styles. The *Manual*, as far as I can see, has little to say on the matter.

> **origpubas:** This type can, if you want, replace the old userf mechanism, described below, for presenting an original with its translation. It's quite similar to the default type, but with a bibstring automatically connecting the entry with its related entries. You can identify other sorts of relationships if you change the introductory string using relatedstring.

> **origpubin:** I have barely altered this from the biblatex default, and it will present reprint information *after* the main entry rather than within it. The *Manual* seems to prefer the latter for the notes & bibliography style and, in some circumstances, the former for author-date.

> **reprintfrom:** This type provides a replacement for the old reprinttitle mechanism described below. As in the standard styles, it presents a fuller reference to the reprinted material than does origpubin, and is designed particularly for presenting pieces formerly printed in other collections or perhaps essays collected from various periodicals. (In biblatex-chicago it contains some kludges to cope with possible babel language environments, so if you find it behaving oddly please let me know, including whether you are using babel [which I've tested] or polyglossia [which I've tested somewhat less].)

Now, the Chicago-specific types:

> **commenton:** I designed the new relatedtype commenton to facilitate citation of online comments, and it is available in two entry types, online and review (with its clone suppperiodical). In both types the *Manual* (15.51–52) recommends that such material appear *only* in the text and not in the reference list, but I have attempted to simplify the presentation of such material wherever you want it to appear. Following the specifications, then, the default when you use commenton is for biblatex-chicago-authordate to modify how your .bib entry appears in the .bbl file by setting both skipbib and cmsdate=full in the options field, so that nothing appears in the reference list and citations present the full date and possibly also a time stamp (see below). Further, the style sets the verbc field so that these entries don't interfere with the provision of extra date letters — the full date and time should be

enough to individuate separate comments. Finally, the style creates a new `customc` entry in your .bbl file which you can cite after your initial `commenton` entry using `\autocites` and which will, as a comment to your initial entry, say whether it's a comment or a reply or what have you, and then giving the short citation of that upon which it is a comment. (Just to be clear: your .bib file will itself never be altered, only the .bbl file, which is produced by `biber` and which provides the data from which `biblatex` actually typesets citations.) Because of how this works, you can currently only use *one* entry key in the `related` field for this `relatedtype`, though I may lift this restriction in the future.

As an example, take the Facebook post diaz:surprise, which does appear in the reference list. The entry licis:diazcomment presents a comment on this post using the `relatedtype` `commenton`, so `biblatex-chicago-authordate` creates a new entry, diaz:surprise-customc. When you cite the comment in your document a command like `\autocites{licis:diazcomment}{diaz:surprise-customc}` will produce a citation like (Licis, February 24, 2016; comment on Díaz 2016). You can alter the string connecting the two citations (by default `\bibstring{commenton}`) by using the `relatedstring` field in the first of them (cf. powell:comment). (Note how minimal the .bib entry of a comment using this system can be — author, related, relatedtype, and `date` are pretty much the only fields required.)

Those who want `online` comments to appear in the reference list can still use the `commenton` `relatedtype`, and the same citation of the commented piece will appear there, connected by the same string that the `customc` entry provides. Here, though, you can also provide a separate title for the comment, and/or a separate url for it, should they exist, which will be printed before/after the citation of the commented piece, respectively. (In `review` entries, which use the same `relatedtype`, only the generic title is available, as is always the case with such entries.) If you manually set either (or both) of the `cmsdate` or the `skipbib` options in your entry then `biblatex-chicago` will assume you want to hand-craft that entry without its intervention, so it won't alter the `options` field or indeed provide any `verbc` field, though it will still provide the virtual `customc` entry in your .bbl file, as that may still prove convenient. Note also that any `verbc` field you provide will never be altered by the package.

reviewof: Philip Kime's biblatex-apa package includes this type, and user Bertold Schweitzer suggested it might be a useful addition to biblatex-chicago, so I've added it to the standard six detailed above. It differs from all of them in that it prints the `relatedstring` (by default `\bibstring{reviewof}`) and the data from the `related` entry in the middle of the parent entry, rather than at the end. It also differs from them in being available only in `article` and `review` entries (along with the latter's clone, `suppperiodical`).

In `article` entries it replaces the `titleaddon` with the `relatedstring` followed by the title of the child entry, and in `review` entries it replaces the `title` with the same two components. In both types these components will optionally be followed by the `author`, `editor`, `translator`, etc., of the reviewed item, and then any child `titleaddon` may optionally appear at the end, allowing maximum flexibility when presenting, for example, reviews of live performances.

This mechanism automates both the provision of the localized `\bibstring` and also the formatting of the `title` of the reviewed work, and it also obviates the need to use any of the `\partedit` macros in this context. If you've changed the default setting of the `related` option in the preamble, then you'll need to ensure that it is set to `true` in the individual entries where you use this `relatedtype` to ensure that the entry's full data appears in the list of references. Also, if the mechanism doesn't work for you in a particular context, remember that the standard way of presenting reviewed works is still available.

short: This `relatedtype` is like the `default` type, only it prints author-date citations rather than full reference-list entries. There is no default `relatedstring` for this type, so if you leave that field blank then the short references will simply appear at the end of the parent entry in the reference list. Any entries you reference in this way

will by default appear separately in the reference list, even if you haven't cited them anywhere else in your document. Otherwise the short reference won't actually be decipherable.

reprinttitle **NB: If you have been using this feature, you may want to have a look at the** related-type reprintfrom, **documented above, for a better solution to this problem, one that also allows you to change the introductory string using the** relatedstring **field. The** reprinttitle **field will continue to work as before, however.** At the request of Will Small, I have included a means of providing the original publication details of an essay or a chapter that you are citing from a subsequent reprint, e.g., a *Collected Essays* volume. In such a case, at least according to the *Manual* (14.181), these details would only appear in the reference list, and then only if these details are "of particular interest." The data would follow an introductory phrase like "originally published as," making the problem strictly parallel to that of including details of a work in the original language alongside the details of its translation. I have addressed the latter problem with the userf field, which provides a sort of cross-referencing method for this purpose, and reprinttitle works in *exactly* the same way. In the .bib entry for the reprint you include a cross-reference to the cite key of the original location using the reprinttitle field (which it may help mnemonically to think of as a "reprinted title" field). The main difference between the two forms is that userf prints all but the author of the original work, whereas reprinttitle suppresses both the author and the title of the original, giving only the more general details, beginning with, e.g., the journaltitle or booktitle and continuing from there. The string prefacing this information will be "Originally published in." Please see the documentation on userf below for all the details on how to create .bib entries for presenting your data.

userf This is one of the supplemental fields which biblatex provides, and is used by biblatex-chicago for a very specific purpose. When you cite both a translation and its original, the *Manual* (14.99) recommends that, in a reference list at least, you combine references to both texts in one entry. Lacking specific instructions about the author-date style, I have nonetheless chosen to implement this possibility also for a list of references, though in-text citations will still only refer to individual works. In order to follow this specification, I have provided a third cross-referencing system (the others being crossref and xref), and have chosen the name userf because it might act as a mnemonic for its function.

In order to use this system, you should start by entering both the original and its translation into your .bib file, just as you normally would. The mechanism works for any entry type, and the two entries need not be of the same type. In the entry for the *translation*, you put the cite key of the original into the userf field. In the *original's* entry, you need to include some means of preventing it appearing separately in the list of references, either a toggle in the keywords field or perhaps skipbib in the options field. In this standard case, the data for the translation will be printed first, followed by the string orig. pub. as, followed by the original, author omitted. As explained above (**origlanguage**), I have also included a way to modify the string printed before the original. In the entry for the *translation*, you put the original's language in origlanguage, and instead of originally published as, you'll get French edition: or Latin edition:, etc. (aristotle:metaphy:gr, aristotle:metaphy:trans). **NB:** You can use the relatedtype origpubas to replicate the userf functionality, and you can also customize the relatedstring field to achieve the same result as with origlanguage.

5.3 Commands

In this section I shall attempt to document all those commands you may need when using biblatex-chicago-authordate that I have either altered with respect to the standard provided by biblatex or that I have provided myself. Some of these, unfortunately, will make your .bib file incompatible with other biblatex styles, but I've been unable to avoid this. Any ideas for more elegant, and more compatible, solutions will be warmly welcomed.

5.3.1 Formatting Commands

These commands allow you to fine-tune the presentation of your references in both citations and list of references. You can find many examples of their usage in dates-test.bib,

and I shall try to point you toward a few such entries in what follows. **NB:** biblatex's `\mkbibquote` command is mandatory in some situations. See its entry below.

\autocap Version 0.8 of biblatex introduced the `\autocap` command, which capitalizes a word inside a citation or list of references entry if that word follows sentence-ending punctuation, and leaves it lowercase otherwise. The whole question of capitalization is considerably more complicated in the notes & bibliography style, where the former uses commas and the latter (often) periods to separate blocks of information, whereas the more streamlined author-date specification has few such issues. In dates-test.bib there are only two places where the `\autocap` macro is necessary, and they both involve the string forthcoming in the year field (author:forthcoming, contrib:contrib), though you can now avoid even this necessity by placing forthcoming in the pubstate field.

I have nonetheless retained the system developed, following biblatex's example, for the notes & bibliography style, which automatically tracks the capitalization of certain fields in your .bib file. I chose these fields after a non-scientific survey of entries in my own databases, so of course if you have ideas for the extension of this facility I would be most interested to hear them. In order to take advantage of this functionality, all you need do is begin the data in the appropriate field with a lowercase letter, e.g., note = {with the assistance of X}. If the data begins with a capital letter — and this is not infrequent — that capital will always be retained. (cf., e.g., creel:house, morgenson:market.) If, on the other hand, you for some reason need such a field always to start with a lowercase letter, then you can try putting an empty set of curly braces {} at the start, which turns off the mechanism without printing anything itself. Here, then, for reference purposes, is the complete list of fields where this functionality is active:

1. The **addendum** field in all entry types.
2. The **booktitleaddon** field in all entry types.
3. The **edition** field in all entry types. (Numerals work as you expect them to here.)
4. The **maintitleaddon** field in all entry types.
5. The **note** field in all entry types.
6. The **part** field in entry types that use it.
7. The **prenote** field prefixed to citation commands.
8. The **relatedstring** field in all entry types.
9. The **shorttitle** field in the review (suppperiodical) entry type and in the misc type, in the latter case, however, only when there is an entrysubtype defined, indicating that the work cited is from an archive.
10. The **title** field in the review (suppperiodical) entry type and in the misc type, in the latter case, however, only when there is an entrysubtype defined, indicating that the work cited is from an archive.
11. The **titleaddon** field in all entry types.
12. The **type** field in artwork, audio, image, music, suppbook, suppcollection, and video entry types.

If you accidentally use the `\autocap` macro in one of the above fields, it really shouldn't matter at all, and you'll still get what you want, but taking advantage of the automatic provisions should at least save some typing.

\bibstring This is a very powerful mechanism to allow biblatex automatically to provide a localized version of a string, and to determine whether that string needs capitalization, depending on where it falls in an entry. Biblatex also provides functionality which allows you sometimes simply to input, for example, newseries instead of `\bibstring{newseries}`, the package auto-detecting when a bibstring is involved and doing the right thing, though in all such cases either form will work. This functionality is available in the series field of article, periodical, and review entries; in the type field of manual, patent, report, and thesis entries; in the location field of patent entries; in the language field in all entry types; and in the nameaddon field in customc entries. These are the places, as far as I can make out, where biblatex's standard styles support this feature, though I have added the last, style-specific, one. If the biblatex authors generalize it still further in a future release, I shall do the same, if possible.

\letterdatelong	I have provided this macro mainly for use in the optional postnote field of the various citation commands. When citing a letter (published or unpublished, letter or misc), it may be useful to include the date in the citation in order to disambiguate references. This macro simply prints the date of a letter, or indeed of any other sort of correspondence, in day-month-year order, as recommended by the *Manual* (14.224). (If your main document language isn't American, it's better just to use the standard biblatex command \printorigdate.)
\mkbibquote	This is the standard biblatex command, which requires attention here because it is a crucial part of the mechanism of that package's "American" punctuation system. Quotation marks around the title field in various entry types are automatically provided by biblatex-chicago, but titles-within-titles frequently also require them, so it is best to get accustomed to using this command to make sure any periods or commas appearing in the neighborhood of the closing quotes will appear inside them automatically. A few examples from dates-test.bib should help to clarify this.

In an article entry, the title contains a quoted phrase:

```
title = {Diethylstilbestrol and Media Coverage of the
    \mkbibquote{Morning After} Pill}
```

Here, because the quoted text doesn't come at the end of title, and no punctuation will ever need to be drawn within the closing quotation mark, you could instead use \enquote{Morning After} or even `Morning After'. (Note the single quotation marks here — the other two methods have the virtue of taking care of nesting for you.) All of these will produce the formatted: "Diethylstilbestrol and Media Coverage of the 'Morning After' Pill."

Here, by contrast, is a book title:

```
title = {Annotations to \mkbibquote{Finnegans Wake}}
```

Because the quoted title within the title comes at the end of the field, and because this reference unit will be separated from what follows by a period in the list of references, then the \mkbibquote command is necessary to bring that period within the final quotation marks, like so: *Annotations to "Finnegans Wake."*

Note in both cases that you only need to be careful with the capitalization inside the curly brackets if you are using authordate-trad, as recent editions of the *Manual* have unified the title formatting for the two remaining styles, which means that, for them, all lower- and uppercase letters remain as they are typed in your .bib file.

Let me also add that this command interacts well with Lehman's csquotes package, which I highly recommend, though the latter isn't strictly necessary in texts using an American style, to which biblatex defaults when csquotes isn't loaded.

\postvolpunct	The *Manual* (14.116) unequivocally prescribes that when a volume number appears immediately before a page number, "the abbreviation *vol.* is omitted and a colon separates the volume number from the page number with no intervening space." The treatment is basically the same whether the citation is of a book or of a periodical, and it appears to be a surprising and unwelcome feature for many users, conflicting as it may do with established typographic traditions in a number of contexts. Clea F. Rees has requested a way to customize this, so I have provided the \postvolpunct command, which prints the punctuation between a volume number and a page number. It is set to \addcolon by default, except when the current language of the entry is French, in which case it defaults to \addcolon\addspace. You can use \renewcommand{\postvolpunct}{...} in your preamble to redefine it, but please note that the command only applies in this limited context, not more generally to the punctuation that appears between, e.g., a volume and a part field.
\partcomp	This and the following 6 macros were all designed to help biblatex-chicago cope with the fact that many bibstrings in the notes & bibliography style differ between notes and bibliography, the former sometimes using abbreviated forms when the latter prints them in

full. These problems do not arise in the author-date styles, but using these macros will make your .bib database more portable across languages and across both Chicago styles, and may be slightly easier to remember than the strings themselves. On the other hand, of course, they will make your .bib file less portable across multiple biblatex styles.

These macros allow you to provide an editor, a translator, and/or a compiler in situations where the available fields (editor, namea, translator, nameb, and namec) aren't adequate. Their names all begin with \part, as originally I intended them for use when a particular name applied only to a specific title, rather than to a maintitle or booktitle (cf. **namea** and **nameb**, above).

In the present instance, you can use \partcomp to identify a compiler when namec (or editortype) won't do, e.g., in a note field or the like. In such a case, biblatex-chicago will print the appropriate string in your references.

\partedit Use this macro when identifying an editor whose name doesn't conveniently fit into the usual fields (editor or namea). (N.B.: If you are writing in French then you no longer need to add either de or d' after this command in your .bib files. The new version of the command should take care of this automatically for you.) See howell:marriage.

\partedit-andcomp As before, but for use when an editor is also a compiler.

\partedit-andtrans As before, but for when when an editor is also a translator (ratliff:review).

\partedit-transandcomp As before, but for when an editor is also a translator and a compiler.

\parttrans-andcomp As before, but for when a translator is also a compiler.

\parttrans As before, but for use when identifying a translator whose name doesn't conveniently fit into the usual fields (translator and nameb).

\reprint This is equivalent to \bibstring{reprint}. It is useful in the notes & bibliography style, and I include it in the author-date styles for compatibility.

\suppress-bibfield[]{} Unlike the other commands presented here, this should be used in your document preamble rather than in your bibliographical apparatus. Also unlike them, it has two arguments, the first of which is optional, the second required. Jan David Hauck suggested that, in addition to the field-exclusion package options provided by biblatex-chicago (see section 5.4.2), I might also provide a general-purpose macro to clear fields from selected entry types when the package options aren't quite right for a user's particular needs. The \suppressbibfield command does this, so that \suppressbibfield{note} clears the note field from *all* entries, while \suppressbibfield[report]{note} clears it only from report entries. Both arguments take comma-separated lists, so to suppress titleaddon and volumes fields from report and manual entries, your preamble could contain \suppressbib-field[report,manual]{titleaddon,volumes}.

A few usage notes are in order. First, you can use as many calls to the command in your preamble as you wish. Second, the command is a very basic user interface to biblatex's source mapping functionality (biblatex.pdf § 4.5.3), so what it does is modify what biber takes from your .bib file in order to produce the .bbl file that biblatex actually reads. As far as biblatex is concerned, the fields simply aren't there in the data source, so they can't appear anywhere in the bibliographical apparatus, whether in citations, reference lists, or shorthand lists. Third, because source mapping is involved, you'll need a complete cycle of LaTeX-biber-LaTeX runs to make the commands take effect. Fourth, source mapping occurs at a very early stage in biber's operation, so if your field names or entry types are standard aliases, the command will only work on the names as they appear in your .bib file, not as they are aliased in the .bbl file. If you have a techreport entry, for example, it won't be affected by a command that alters report entries, and a date field won't be affected by a command that suppresses the year. Fifth, the code for the command resides in biblatex-chicago.sty, so if you load the styles without loading that package it won't be

available to you. Sixth and finally, the \suppressbibfield command is new and relatively untested, so please report any untoward behavior to me.

5.3.2 Citation Commands

The biblatex package is particularly rich in citation commands, most of which, in biblatex-chicago-authordate and authordate-trad, function as they do in the standard author-date styles. If you are getting unexpected behavior when using them please have a look in your .log file. A command like \supercite, listed in § 3.6.2 of the biblatex manual but not defined by biblatex-chicago-authordate or by core biblatex, defaults to \cite, and leaves a warning in the .log. The following commands may require some minimal explanation, but if there are standard commands that don't work for you, or new commands that would be useful, please let me know, and it should be possible to fix or add them.

\atcite
\atpcite These two new citation commands allow you quickly and easily to provide an author-title citation of any entry, instead of an author-date citation. The classical entrysubtype field does this, but it also changes punctuation in the citation, so I've provided other means to achieve the same end. The first of these new citation commands presents the plain citation, the second includes it in parentheses for inclusion in running text. The new authortitle type and entry option (section 5.4.3) has the same effect when using the standard citation commands, but it's possible that using these new commands instead may give added flexibility.

\autocite I haven't adapted this in the slightest, but I thought it worth pointing out that biblatex-chicago-authordate sets this command to use \parencite as the default option. It is, in my experience, much the most common citation command you will use, and also works fine in its multicite form, **\autocites**.

\citeincite Although the officially-sanctioned, and safest, way to present cross-references to other works is by using the related mechanism, it would appear, judging from various bug reports I've received, that there are plenty of users who need to present such references in ways that aren't supported by the related code as it stands. This new citation command is designed to allow you to present short references to other works inside other references, and to avoid some, I hope most, of the bugs associated with using cite commands in the fields of other entries. If you want to present long citations, I'm assuming that you wouldn't want to do that in fields right in the middle of the parent entry, and that, therefore, the usual methods detailed in section 5.2.1 (above) or the \fullciteincite command (below) will serve your needs. Similarly, if you want to provide short citations at the end of your parent entry, the new relatedtype short should work.

If, however, you must have a short reference in the midst of another field, then \citeincite can help. It is certified to work properly only in the addendum, annotation, annote, note, and titleaddon fields, and in the author-date styles also in the title fields of customc entries, which allows you (via a multicite command) to include such citations as comments to other in-text citations. The multicite command \citeincites is also available in the same contexts. In previous releases of biblatex there were differences in how the case-changing backends behaved with respect to citation commands inside entry fields, but in my testing the two backends appear currently to be equivalent on this score. Still, should you run into problems using \citeincite, it might still be worth setting case-changer=latex2e in your preamble to see if that helps.

\fullciteincite Ordinarily, you can use one of the methods discussed in section 5.2.1 to present, in a reference list, a full citation of a work related in some way to the current entry. The new command \citeincite (above) allows you to present short citations to other works within selected fields of the parent entry, or indeed as part of comments inside other in-text citations, while attempting to minimize the problems caused by citations within other citations. The \fullciteincite command works similarly for full citations, but really is designed for use only in annotation fields, the idea being that you might want to refer in an annotated reference list to works that you haven't cited in the main body of your text. The multicite command \fullciteincites is also available. It may be worth noting

here that, because the full reference list drivers don't contain code for printing postnote fields, any postnotes that you provide will be silently ignored. Inconsistently enough, prenote, multiprenote, and multipostnote fields will appear.

\gentextcite   Arne Skjærholt requested, for the author-date styles, a variant of the \textcite command that presented the author's name in the genitive case in running text, thereby simplifying certain syntactic constructions (15.25). The \gentextcite command, in effect, provides a way to include almost anything in between the name and the parenthesized date in a \textcite, so its use may well not be limited to the possessive. In most respects it behaves exactly like \textcite, on which see below. The difference is that I've added a new optional field to the front of the command to allow you to choose which declensional ending to add to the name. If you don't specify this field, you'll get the standard English "'s". If you want something different, you'll need to present a third option to the command, like so: \gentextcite[<ending>][][]{entry:key}. You must include the two further sets of square brackets, because with only one set it will, as with other citation commands, be interpreted as a postnote, and with two a prenote and a postnote. There is a \gentextcites command as well, though currently you can only specify one genitival ending for all keys, like so: \gentextcites[<ending>]()()[][]{entry:key1}{entry:key2}, though if you don't have a pre- or postnote to the first citation you can make do with \gentextcites[<ending>](){entry:key1}{entry:key2}.

The syntax of multiple authors' names in running text is unpredictable. There is currently no way to add the genitival ending to all the names attached to a single citation key, so it will only appear at the end of a group of names in such a case. (This is in keeping with the usual syntax when referring to a multi-author work, at least in English.) When using \gentextcites, however, you can control whether the ending appears after the name(s) attached to each citation key, or whether it only appears after the names attached to the last key. By default, it only appears after the last, but the genallnames preamble and/or entry option set to true will attach the ending to each key's name(s). When using one citation command to cite more than one work by the same author, it is the *first* occurrence of the name which biblatex prints, eliding subsequent ones. In order to get the possessive ending on that name you'll need to set genallnames to true.

\textcite   In standard biblatex this command searches first for a labelname, usually taken from the author or shortauthor field, then uses the shorthand field if the former doesn't exist. Because of the way the Chicago author-date specification recommends handling abbreviations, I have switched this around, and the command searches for a shorthand first. This holds also for the multicite form **\textcites**, though both commands revert to their standard biblatex behavior when you give the cmslos=false option in the preamble.

5.4 Package Options

5.4.1 Pre-set biblatex Options

Although a quick glance through biblatex-chicago.sty will tell you which biblatex options the package sets for you, I thought I might gather them here also for your perusal. These settings are, I believe, consistent with the specification, but you can alter them in the options to biblatex-chicago in your preamble or by loading the package using \usepackage [style=chicago-authordate]{biblatex}, which gives you the biblatex defaults unless you redefine them yourself inside the square brackets.

autocite=   Biblatex-chicago-authordate and authordate-trad place references in parentheses by de-
inline   fault.

citetracker=   The citetracker for the \ifciteseen test is enabled globally.
true
alldates=comp   The specification calls for the long format when presenting dates, slightly shortened when presenting date ranges. The new style option compressyears, enabled by default, means that year ranges are now compressed automatically according to the *Manual's* instructions (9.64; section 5.4.2).

alltimes=12h	In entries which print time stamps, they will, when the stamp is part of a date, eventdate, or origdate, appear in 12-hour format, i.e., "4:45 p.m." Stamps that are part of a urldate are, by default, controlled by the urltime option, which is set to 24h. See that option below, and table 3.
datecirca=true	This option enables biblatex's enhanced "circa" date specification, which given a date like 1989~ will print [ca. 1989]. Cf. table 3.
dateuncertain=true	This option enables biblatex's enhanced "uncertain" date specification, which given a date like 1989? will print [1989?]. A field like 1989% is both "circa" *and* "uncertain," like so: [ca. 1989?]. Cf. table 3.
datezeros=false	This ensures that leading zeros don't appear in date specifications.
ibidtracker=constrict	This enables an *ibidem* mechanism in citations, but only in the most strictly-defined circumstances. The Chicago author-date style doesn't print "Ibid" in citations, but in general a repeated citation on the same page will print only the page reference. Technically, this should only occur when a source is cited "more than once in one paragraph" (15.27), so you can use the \citereset command from biblatex to achieve the greatest compliance, as the package only offers automatic resetting on part, chapter, section, and subsection boundaries, while biblatex-chicago automatically resets the tracker at page breaks. (Cf. biblatex.pdf § 3.1.2.1.) Whenever there might be any ambiguity, biblatex should default to printing a more informative reference.
	If you are going to repeat a source, make sure that the cite command provides a postnote — you'll no longer get any annoying empty parentheses, but you will get another standard citation, which may add too much clutter.
labeldateparts=true	This option tells biblatex to provide the special labelyear and extradate fields for author-date styles. (This is the option formerly known as labelyear and then labeldate, both of which are obsolete.)
maxbibnames=10 minbibnames=7	These two options control the number of names printed in the list of references when that number exceeds 10. These numbers follow the recommendations of the *Manual* (14.76, 15.9), and they are different from those for use in citations. Please see section 5.5.2 below (and the file cms-dates-intro.pdf) for hints on dealing with entries with more than three authors.
pagetracker=true	This enables page tracking for the \iffirstonpage and \ifsamepage commands for controlling, among other things, the *ibidem* mechanism. It tracks individual pages if LaTeX is in oneside mode, or whole spreads in twoside mode.
punctfont=true	This fixes a minor problem with punctuation in titles, ensuring that the colon between a title and a subtitle appears in the correct, matching font.
related=true	This is the standard biblatex bibliography option, and it enables the use of related functionality in the list of references. I have added an entry option, as well, so if you set this to false in your preamble, in the options field, or in the relatedoptions field, you can make the package ignore the related fields.
sortcase=false	This turns off the sorting of uppercase and lowercase letters separately, a practice which the *Manual* doesn't appear to recommend.
sorting=cms	This setting takes advantage of the \DeclareSortingTemplate command provided by biblatex and Biber, in effect implementing a default sorting order in the list of references tailored to comply with the Chicago author-date specification. Please see the documentation of \DeclareSortingTemplate in section 5.4.1, below.
timezones=true	If you provide a timezone for a time stamp, usually using one of the timezone fields, this option ensures it will be printed.
uniquelist=minyear	This option enables biblatex-chicago-authordate to disambiguate entries which have more than three authors, but which differ *after* the first name in the list. This will only occur when two such entries have the same year (15.29). The option is Biber-only, like the following, which means that this next-generation BIBTeX replacement is required for the author-date styles. Please see cms-dates-intro.pdf and section 5.5.2, below, for further details.

uniquename= minfull	This enables the package to distinguish different authors who share a surname, using initials in the first instance, and whole names if initials aren't enough (15.22). The option is Biber-only, like the previous one.
urltime=24h	In entries with urldate fields containing time stamps, that stamp will by default appear in 24-hour format, i.e., "16:45." Cf. alltimes, above, urlstamp in section 5.4.2 below, and table 3.
[standard] useeditor=false usenamec=false	In standard entries any editors' or compilers' names appear after the title, according to 14.259, so these entry-type-specific options encode this. You can, of course, override these defaults in your preamble, should you deem it necessary.
usetranslator =true	This enables automatic use of the translator at the head of entries in the absence of an author or an editor. In the list of references, the entry will be alphabetized by the translator's surname. You can disable this functionality on a per-entry basis by setting usetranslator=false in the options field. Cf. silver:gawain.

Other biblatex Formatting Options

I've chosen defaults for many of the general formatting commands provided by biblatex, including the vertical space between items in the list of references and between items in the list of shorthands (\bibitemsep and \lositemsep). I define many of these in biblatex-chicago.sty, and of course you may want to redefine them to your own needs and tastes. It may be as well you know that the *Manual* does state a preference for two of the formatting options I've implemented by default: the 3-em dash as a replacement for repeated names in the list of references (15.17–19, and just below); and the formatting of note numbers, both in the main text and at the bottom of the page / end of the essay (superscript in the text, in-line in the notes; 14.24). The code for this last formatting is also in biblatex-chicago.sty, and I've wrapped it in a test that disables it if you are using the memoir class, which I believe has its own commands for defining these parameters. You can also disable it by using the footmarkoff package option, on which see below.

Gildas Hamel pointed out that my default definition, in biblatex-chicago.sty, of biblatex's \bibnamedash didn't work well with many fonts, leaving a line of three dashes separated by gaps. He suggested an alternative, which I've adopted, with a minor tweak to make the dash thicker, though you can toy with all the parameters to find what looks right with your chosen font. The default definition is: \renewcommand*{\bibnamedash}{\rule[.4ex]{3em}{.6pt}}.

losnotes & losendnotes	At the request of Kenneth Pearce, I have added two bibenvironments to chicago-author-date.bbx, for use with the env option to the \printshorthands command. The first, losnotes, is designed to allow a list of shorthands to appear inside footnotes, while losendnotes does the same for endnotes. Their main effect is to change the font size, and in the latter case to clear up some spurious punctuation and white space that I see on my system when using endnotes. (You'll probably also want to use the option heading=none in order to get rid of the [oversized] default, providing your own within the \footnote command.) If you use a command like \printbiblist{shortjournal} to print a list of journal abbreviations, you can use the sjnotes and sjendnotes bibenvironments in exactly the same way. Please see the documentation of shorthand and shortjournal in section 5.2 above for further options available to you for presenting and formatting these two types of biblist.

The next-generation backend Biber and biblatex offer enhanced functionality in many areas, including the next three declarations. If the default definitions don't work well for you, you can redefine all of them in your document preamble — see biblatex.pdf §§ 4.5.6 and 4.5.10.

\Declare-Labelname	This option allows you to add name fields for consideration when biblatex is attempting to find a shortened name for in-text citations. This, for example, allows a compiler (=namec) to appear in citations without any other intervention from the user, rather than requiring a shortauthor field as previous releases of biblatex-chicago did. The default definition currently is {shortauthor,author,shorteditor,namea,editor,nameb,translator,namec}.
\Declare-Labeldate	This option allows you to alter the order in which Biber and biblatex search for the year to use both in citations and at the head of entries in the list of references. This will also be

the year to which an alphabetical suffix will be appended when an author has published more than one work in the same year, and the year by which works will be sorted in the list of references. In the default configuration, a year will be searched for in the order date, eventdate, origdate, urldate. This generally suits the Chicago author-date styles well, except for two situations. First, when a reference apparatus contains many entries with multiple dates, it may be simplest to promote the origdate to the head of the list, which you can do using the cmsdate preamble option. This changes the order to origdate, date, eventdate, urldate. Second, in music and video entries, and, exceptionally, some review entries, the general rule is to emphasize the earliest date. For these three entry types, then, \DeclareLabeldate uses the order eventdate, origdate, date, urldate. See avdate in section 5.4.2, cmsdate in section 5.4.3, and the **date** docs in section 5.2.

\Declare-Sorting-Template — The third Biber enhancement I have implemented allows you to include almost any field whatsoever in biblatex's sorting algorithms for the list of references, so that a great many more entries will be sorted correctly automatically rather than requiring manual intervention in the form of a sortkey field or the like. Code in biblatex-chicago.sty sets the biblatex option sorting=cms, which is a custom scheme, basically a Chicago-specific variant of the default nyt. You can find its definition in chicago-authordate.cbx. (Please note that it uses the labelyear as its main year component, which should help improve the automatic sorting of entries by the same author.)

The advantages of this scheme are, specifically, that any entry headed by one of the supplemental name fields (name[a-c]), a manual or a standard entry headed by an organization, or an article or review entry with an entrysubtype and headed by a journaltitle will no longer need a sortkey set. Further, the biblatex use<name>=false options will remove any name field from the sorting order, again reducing the need for user intervention. The main disadvantage should only occur very rarely. In author-less article and review entries without an entrysubtype, the title will appear instead of the journaltitle, and since the latter appears before the former in the sorting scheme, you'll need a sortkey for proper alphabetization.

5.4.2 Pre-set chicago Options

bookpages=true — At the request of Scot Becker, I have included this rather specialized option, which controls the printing of the pages field in book entries. Some bibliographic managers, apparently, place the total page count in that field by default, and this option allows you to stop the printing of this information in the reference list. It defaults to true, which means the field is printed, but it can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field (though rather than use this latter method it would make sense to eliminate the pages field from the affected entries).

doi=true — This option controls whether any doi fields present in the .bib file will be printed in the reference list. At the request of Daniel Possenriede, and keeping in mind the *Manual's* preference for this field instead of a url (14.7–8), I have added a third switch, only, which prints the doi if it is present and the url only if there is no doi. Ryo Furue more recently requested a way to suppress the urldate when using only the doi, so I've added the onlynd switch to do this. The package default remains the same, however — it defaults to true, which will print both doi and url if both are present. The option can be set to only, onlynd, or to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field. In online entries, the doi field will always be printed, but the only switch will still eliminate any url, and onlynd will still eliminate both the url and the urldate.

eprint=true — This option controls whether any eprint fields present in the .bib file will be printed in the list of references. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis, in the options field. In online entries, the eprint field will always be printed.

isbn=true — This option controls whether any isan, isbn, ismn, isrn, issn, and iswc fields present in the .bib file will be printed in the list of references. It defaults to true, and can be set to

	false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis, in the options field.
numbermonth =true	Once again at the request of Scot Becker, I have included this option, which controls the printing of the month field in all the periodical-type entries when a number field is also present. Some bibliographic software, apparently, always includes the month of publication even when a number is present. When all this information is available the *Manual* (14.171) prints everything, so this option defaults to true, which means the field is printed, but it can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field.
url=true	This option controls whether any url fields present in the .bib file will be printed in the reference list. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis, in the options field. Please note that, as in standard biblatex, the url field is always printed in online entries, regardless of the state of this option.
urlstamp=true	This option controls whether any urltime fields, included as part of the urldate, will be printed in citations and reference list. It defaults to true, and can be set to false either in the preamble, for the whole document or for specific entry types, or on a per-entry basis in the options field. Please note that, unlike the url option, this option *does* control what is printed in online entries.
includeall= true	This is the one option that rules the seven preceding, either printing all the fields under consideration — the default — or excluding all of them. It is set to true in chicago-authordate.cbx, but you can change it either in the preamble for the whole document or for specific entry types, or in the options field of individual entries. The seven individual options above are similarly available in the same places, for finer-grained control. The rationale for all of these options is the availability of bibliographic managers that helpfully present as much data as possible, in every entry, some of which may not be felt to be entirely necessary. Setting includeall to true probably works just fine for those compiling their .bib databases by hand, but others may find that some automatic pruning helps clear things up, at least to a first approximation. Some per-type or per-entry work afterward may then polish up the details. If you find that you need control over fields that aren't included among these options, I have provided the \suppressbibfield command for your preamble, as suggested by Jan David Hauck. It is in fact a user interface to the source mapping feature of biblatex, and it is something of a nuclear option, preventing fields from even appearing in the .bbl file generated by biber from your .bib database. See the \suppressbibfield command in section 5.3.1 and the source mapping docs in biblatex.pdf § 4.5.3.
avdate=true	For music and video entries, the *Manual* (14.263, 15.57) strongly recommends both that you provide a recording, release, or broadcast date for your references and also that this earlier date should appear in citations and at the head of reference list entries. In the default setting of \DeclareLabeldate, biblatex searches for dates in the following order: year, eventyear, origyear, urlyear. This option changes the default ordering in music and video entries to the following: eventyear, origyear, year, urlyear. Review entries presenting on-line comments have similar needs, as do standard entries, so the same reordering applies to those entry types, too. If you simply want to apply the defaults to these four entry types, you can use avdate=false in the options when loading biblatex-chicago. If, however, you want to tailor the algorithm to your own needs, then you can use \DeclareLabeldate commands in your preamble. Please be aware, however, that some parts of the style hard-code the search syntax, and although they take account of the avdate setting, if you use your own definitions of \DeclareLabeldate the results may, in some corner cases, surprise. Please see music, review, standard, and video in section 5.1; date, eventdate, origdate, and urldate in section 5.2; and \DeclareLabeldate in section 5.4.1.
bibannotesep =vpar citeannotesep =period	These options define the relation of the annotation field to the main entry, bibannotesep doing so in the reference list and citeannotesep in long (legal) notes. (The annotation option in section 5.4.3 determines where, if anywhere, the field will appear.) Both options have the same set of keys, though they have different default settings if you don't define them yourself. The possible values are:

none = no punctuation at all.

space = \addspace

comma = \addcomma\addspace

period = \addperiod\addspace

colon = \addcolon\addspace

semicolon = \addsemicolon\addspace

par: This starts a new paragraph on the next line. Page breaking is strongly inhibited before the annotation.

vpar: This starts a new paragraph, and also inserts some vertical space before it. In long notes that vertical space is minimal (1 pt), while in the reference list it creates a blank line. Page breaking is strongly inhibited before the annotation.

parbreak: This is the same as par, but it allows a page break to occur between the main entry and the start of the annotation.

vparbreak: This is the same as vpar, but it also allows a page break between the main entry and the annotation.

Please note that both of these options are available in the preamble both globally and per-type, and also in the options field of individual entries. Each defines a command (\bibannotesep and \citeannotesep) which appears in the annotation field's formatting directive, so it's possible to redefine these commands in your preamble if you have needs that the available values don't address. (You can also try sending an email to encourage me to add other keys.) Please also keep in mind that bibannotesep interacts with the entrybreak and formatbib options in section 5.4.3, below, to determine the general layout of the reference list. Depending on the settings of those options, changing the bibannotesep from entry to entry may not work out well.

booklongxref=
true
At the request of Bertold Schweitzer, I have included two options for controlling whether and where biblatex-chicago will print abbreviated references when you cite more than one part of a given collection or series. This option controls whether multiple book, bookinbook, collection, and proceedings entries which are part of the same collection will appear in this space-saving format. The parent collection itself will usually be presented in, e.g., a book, bookinbook, mvbook, mvcollection, or mvproceedings entry, and using crossref or xref in the child entries will allow such presentation depending on the value of the option:

true: This is the default. If you use crossref or xref fields in these entry types, by default you will *not* get any abbreviated citations in the reference list.

false: You'll get abbreviated citations in these entry types in the reference list.

notes, bib: These two options are carried over from the notes & bibliography style; here they are synonymous with false and true, respectively.

This option can be set either in the preamble or in the options field of individual entries. For controlling the behavior of inbook, incollection, inproceedings, and letter entries, please see longcrossref, below, and also the documentation of crossref in section 5.2.

cmslos=true
This option alters biblatex's standard behavior when processing the shorthand field. Chicago's author-date style only seems to recommend the use of shorthands as abbreviations for long authors' names, particularly institutional names, which means the shorthand will replace only the name part in citations rather than the whole citation (15.37; bsi:abbreviation, iso:electrodoc). Recent editions suggest placing the abbreviation at the head of the entry, followed by its expansion inside parentheses, an arrangement automatically provided by biblatex-chicago-authordate when you use the shorthand field, assuming you retain the default setting of this option. Please note that you can still print a list of shorthands if you wish, and you can also get back something approaching the "standard" behavior of shorthands if you give the cmslos=false option to biblatex-chicago in your document preamble. Cf. section 5.2, s.v. "**shorthand**" above, and also cms-dates-intro.pdf.

compressyears
=true
The *Manual* has long recommended (9.64, 15.41), as a space-saving measure, the compression of year ranges when presenting dates. I have, finally, implemented this in the

current release, and have made it the default, which you can change in your document preamble. Please note that the rules for compressing years are different from those for compressing other numbers (e.g., page numbers), and also that the compression code is in biblatex-chicago.sty, which will have to be loaded for this option to make any difference. Cf. table 3.

ctitleaddon= comma
ptitleaddon= period

Roger Hart requested a way to control the punctuation printed before the titleaddon, booktitleaddon, and maintitleaddon fields. By default, this is \addcomma\addspace (ctitleaddon) for nearly all book- and maintitleaddons in the list of references, while \addperiod\addspace (ptitleaddon) is the default before most titleaddons there. If the punctuation printed isn't correct for your needs, you can set the relevant option either in the preamble or in individual entries. (Cf. coolidge:speech and schubert:muellerin.) The accepted option keys are:

none = no punctuation at all
space = \addspace
comma = \addcomma\addspace
period = \addperiod\addspace
colon = \addcolon\addspace
semicolon = \addsemicolon\addspace

If you need something a little more exotic, you can directly \renewcommand either \ctitleaddonpunct or \ptitleaddonpunct (or both) in your preamble, but it's worth remembering that the redefinition will hold for all instances, unless you use the options field in your other entries with a titleaddon field. A simpler solution might be to set the relevant option to none in your entry and then include the punctuation in the titleaddon field itself.

dashed=true

Constanza Cordoni has requested a way to turn off the 3-em dash for replacing repeated names in the reference list, and the *Manual* admits that some publishers prefer this, as the dash can carry with it certain inconveniences, especially for electronic formats (15.17). Some of biblatex's standard styles have a dashed option, so for compatibility purposes I've provided the same. By default I have set it to print the name dash, but you can set dashed=false globally, per type, or per entry to repeat names as and when required.

hidevolumes= true

If both a volume and a volumes field are present, as may occur particularly in cross-referenced entries, then biblatex-chicago will ordinarily suppress the volumes field. In some instances, when a maintitle is present, this may not be the desired result. In this latter case, if the volume appears before the maintitle, this option, set to true by default, controls whether to print the volumes field after that title or not. Set it to false either in the preamble or in the options field of your entry to have it appear after the maintitle.

journalabbrev =notes

This option controls the printing of the shortjournal field in place of the journaltitle field in citations and reference lists. It is set to notes by default, so as shipped biblatex-chicago-authordate will print such fields only in citations, but you can set it, either in the preamble or in individual entries, to one of three other values: true prints the abbreviated form both in citations and reference lists, bib in the reference list only, and false in neither. Please note that in periodical entries the title and shorttitle fields behave in exactly the same manner. For more details, see the documentation of **shortjournal** in section 5.2, above.

jtitleaddon= space

I have added the standard biblatex journaltitleaddon field to the article and review entry types, and also the titleaddon field to the periodical type, fields that may, for example, be particularly useful when you want to provide the original form of a translated journal title. The jtitleaddon option controls the separator between the main title and the addon, as with the ctitleaddon and ptitleaddon options, above, and like them is settable globally, per type, or per entry. The possible settings are the same as for those options, but the default is a space. You can redefine \jtitleaddonpunct directly if you have more unusual needs.

longcrossref= false

This is the second option, requested by Bertold Schweitzer, for controlling whether and where biblatex-chicago will print abbreviated references when you cite more than one

part of a given collection or series. It controls the settings for the entry types more-or-less authorized by the *Manual*, i.e., inbook, incollection, inproceedings, and letter. The mechanism itself is enabled by multiple crossref or xref references to the same parent, whether that be, e.g., a collection, an mvcollection, a proceedings, or an mvproceedings entry. Given these multiple cross references, the presentation in the reference apparatus will be governed by the following options:

false: This is the default. If you use crossref or xref fields in the four mentioned entry types, you'll get the abbreviated entries in the reference list.

true: You'll get no abbreviated citations of these entry types in the reference list.

none: This switch is special, allowing you with one setting to provide abbreviated citations not just of the four entry types mentioned but also of book, bookinbook, collection, and proceedings entries.

notes, bib: These two options are carried over from the notes & bibliography style; here they are synonymous with false and true, respectively.

This option can be set either in the preamble or in the options field of individual entries. For controlling the behavior of book, bookinbook, collection, and proceedings entries, please see booklongxref, above, and also the documentation of crossref in section 5.2.

nameaddonsep
=space

This option sets the punctuation which appears before the nameaddon field in all entry types except customc. You can set it globally, per type, or per entry, using one of the six following keys:

space = \addspace. This is the default.
none = no separator at all. It presumes that you will include one in the nameaddon field itself.
colon = \addcolon\addspace.
comma = \addcomma\addspace.
period = \addperiod\addspace.
semicolon = \addsemicolon\addspace.

Cf. nameaddon and nameaddonformat in section 5.4.3.

nodates=true

This option means that for all entry types except misc and dataset biblatex-chicago will automatically provide \bibstring{nodate} for any entry that doesn't otherwise provide a date for citations and for the heads of entries in the list of references. If you set nodates=false either in your preamble (for global or for per-type coverage) or in individual entries then the package won't perform this substitution. (The bibstring expands to "n.d." in English.)

5.4.3 Style Options – Preamble

These are parts of the specification that not everyone will wish to enable. All except the sixth, ninth, and thirteenth can be used even if you load the package in the old way via a call to biblatex, but most users can just place the appropriate string(s) in the options to the \usepackage{biblatex-chicago} call in your preamble.

alwaysrange

Biblatex-chicago now implements biblatex's enhanced date specifications, one part of which is the presentation of decades and centuries not as year ranges but as localized strings like "19th c." or "1970s." The alwaysrange option set to true, either in your preamble or in individual entries, simply tells the package to present the year range instead. This allows you to use the efficient enhanced notations in the date field ({18XX} or {197X}) without the localized strings appearing, should you require it. The two options centuryrange and decaderange limit the same effect to centuries and decades, respectively. Please see table 3.

annotation

At the request of Emil Salim, I included in biblatex-chicago the ability to produce annotated reference lists. More recently, Moritz Wemheuer brought to my attention a Stack-Exchange question which suggested that the field might be useful in several other contexts as well, so I've modified the annotation option to allow the field to appear in the

reference list (=bib or =true, the default, if no string is given), in long (legal) notes (=notes), in both (=all), or in neither (=false). You can now set the option in the preamble both globally and per-type, and in the options field of individual entries. There are two new options (bibannotesep and citeannotesep) to allow you to choose the separator between the annotation and the rest of the entry, and also two new options (formatbib and entrybreak) to give you fine-grained control over the presentation of the reference list as a whole, including an annotated one. Please have a look at the documentation for the latter two options just below, for the former two options in section 5.4.2, and for the annotation field in section 5.2. Please also see biblatex.pdf § 3.13.8 for details on how to use external files to store annotations.

authortitle
In a few contexts — classical references, some archival material, perhaps scientific databases — the provision of a date for citations may well be impossible, irrelevant, or both. The entrysubtype value classical results in author-title citations for the entry containing it, but it modifies punctuation in those citations in a way that might be wrong for some sources, and it's also possible that you may need the entrysubtype field for some other purpose yet still wish to present author-title citations. You can set authortitle to true either for a specific entry type in the preamble or in the options field of individual entries to achieve this. You can also use the citation commands \atcite and \atpcite, instead, if that's more convenient. The shorttitle field provides a way to abbreviate long titles in this context. Please note that biblatex-chicago by default sets this to true for the dataset entry type, so you can set it to false if you want to present such entries differently.

casechanger
Like mcite and natbib, this is a standard biblatex option which biblatex-chicago simply passes through to that package. In biblatex it defaults to auto, but there were, and possibly still are, cases when the old latex2e case-changing code can work around some bugs when using, for example, citation commands inside fields that have a case-changing element as an automatic part of their formatting (note, titleaddon, type). Cf. section 3.1.1 of biblatex.pdf.

centuryrange
This option works just like alwaysrange, above, but only affects century presentation, not decade. Cf. table 3.

cmsbreakurl
The *Manual* gives fairly specific instructions about breaking URLs across lines (14.18), so I have attempted to implement them by tweaking biblatex's default settings, which are found in biblatex.def. In truth, I haven't succeeded in getting biblatex flawlessly to follow all of the *Manual's* instructions, nor do the changes I have made work well in all circumstances, being particularly unsightly if you happen to be using the ragged2e package. For these reasons, I have made my changes dependent on a package option, cmsbreakurl, which you can set in your preamble. I have placed all of this code in biblatex-chicago.sty, so if you load the package with a call to biblatex instead, then URL line breaking will revert to the biblatex defaults. See cms-dates-sample.pdf for a lot of examples of what URLs look like when the option is set, and also section 5.5.1, below.

cmsdate
This option used *in the preamble* provides a method for simplifying the creation of databases with a great many multi-date entries. Despite warnings in previous releases, users have plainly already been setting this option in their preambles, so I thought I might at least attempt to make it work as "correctly" as I can. The switches for it are basically the same as for the entry-only option, that is, assuming an entry presents a reprinted edition of a work by Smith, first published in 1926 (the origdate) and reprinted in 1985 (the date):

1. cmsdate=off is the default: (Smith 1985).

2. cmsdate=both prints both the origdate and the date, using the *Manual's* standard format: (Author [1926] 1985) in parenthetical citations, Author (1926) 1985 outside parentheses, e.g., in the reference list.

3. cmsdate=on prints the origdate at the head of the entry in the list of references and in citations: (Author 1926). NB: The *Manual* no longer includes this among the approved options. If you want to present the origdate at the head of an entry, then generally speaking you should probably use cmsdate=both. I have nevertheless retained this option for certain cases where it has proved useful. The old options new and old work like both.

The important information for the user is that, when you set this option in your preamble to on or both (or to the old synonyms for the latter, new or old), then biblatex-chicago-authordate (and authordate-trad) will change the default \DeclareLabeldate definition so that the labelyear search order will be origdate, date, eventdate, urldate. This means that for entry types not covered by the avdate option, and for those types as well if you turn off that option, the labelyear will, in any entry containing an origdate, be that very date. If you want *every* such entry to present its origdate in citations and at the head of reference list entries, then setting the option this way makes sense, as you should automatically get the proper extradate letter (1926**a**) and the correct sorting, without having to use the counter-intuitive .bib file date switching that sometimes accompanied the entry-only cmsdate option. A few clarifications may yet be in order.

Obviously, any entry with only a date should behave as usual. Also, since patent entries have fairly specialized needs, I have exempted them from this change to \DeclareLabeldate. Third, the per-entry cmsdate options will still affect which dates are printed in citations and at the head of reference list entries, but they cannot change the search order for the labeldate. This will be fixed by the preamble option. Fourth, if you have been used to switching the date and the origdate to get the correct results, then you should be aware that this mechanism may actually still be useful when using the on switch to cmsdate in the preamble, but it produces incorrect results when the cmsdate option is both in the preamble and the individual entry. The preamble option is designed to make the need for this switching as rare as possible, so some editing of existing databases may be necessary. Fifth, the entry-only option full has no effect at all when used in the preamble; you must set it in individual entries. Finally, please see the documentation of the **date** field in section 5.2 for the fullest discussion of date presentation in the authordate styles.

cmsorigdate   Although I can't currently think of any reason why anyone would want to use it on its own, I should nonetheless mention that the cmsorigdate option in your preamble will change the default \DeclareLabeldate settings to origdate, date, eventdate, urldate. Setting cmsdate to on or both in the preamble — see the previous option — sets this to true, but if for some reason you want to set it to true without any of the other effects of the cmsdate option, then you can. The effects may surprise.

compresspages   When set to true, any page ranges in your .bib file or in the postnote field of your citation commands will be compressed in accordance with the *Manual's* specifications (9.61). Something like 321--328 in your .bib file would become 321–28 in your document. See the pages field in section 5.2, above. Please note that the code for this is in biblatex-chicago.sty, so if you load the package with a call to biblatex instead then you'll get the default biblatex compression style.

datamodel   This is the standard biblatex option for loading the named data model file (excluding its .dbx extension). After a request by Philipp Immel, you can now set this option when you load the Chicago styles with \usepackage{biblatex-chicago}, and it will be passed through properly to biblatex itself. Cf. biblatex.pdf § 4.5.4.

decaderange   This option works just like alwaysrange, above, but only affects decade presentation, not century. Cf. table 3.

entrybreak   This option only makes sense when used in conjunction with the annotenp switch to the formatbib option (below). The latter allows LaTeX to break a page inside an annotation field printed without starting a new paragraph, and it does so by allowing such a break in the entry only after a set number of lines, by default set to 3. The idea is that most reference list entries will fit within 3 lines, so the break would generally be somewhere inside the annotation. If your document needs a value different from 3, provide the integer using the entrybreak option in your preamble. Some experimentation may be needed to find the optimum number for a given document.

footmarkoff   Although the *Manual* (14.19) recommends specific formatting for footnote (and endnote) marks, i.e., superscript in the text and in-line in foot- or endnotes, Charles Schaum has brought it to my attention that not all publishers follow this practice, even when requiring Chicago style. I have retained this formatting as the default setup, but if you include the footmarkoff option, biblatex-chicago will not alter LaTeX's (or the endnote package's) defaults in any way, leaving you free to follow the specifications of your publisher. I have

placed all of this code in `biblatex-chicago.sty`, so if you load the package with a call to biblatex instead, then once again footnote marks will revert to the LaTeX default, but of course you also lose a fair amount of other formatting, as well. See section 5.5.1, below.

formatbib   The *Manual* in fact says very little about formatting issues in reference lists, e.g., whether to break entries across pages and whether to allow widows and orphans (single lines at the start or end of a page). A quick and non-scientific survey of publications issued by the University of Chicago Press suggests that actual practices are extremely varied, so I've tried to provide a number of choices for users of biblatex-chicago, most of them available as keys to the formatbib preamble option, but a few of them also involving settings of the bibannotesep and the entrybreak options. The keys of formatbib are as follows:

> **max:** This is the biblatex default, so if you don't set formatbib at all it's what you get. It provides maximal intervention, disallowing entries broken across pages, including even when an entry includes a lengthy annotation.
> **min:** This allows page breaks just about anywhere, including inside entries, and it also allows widows and orphans, so it will usually provide the most efficient use of available space on the pages of your bibliography.
> **minwo:** This is like min, but discourages widows and orphans.
> **annote:** This option treats annotations separately from reference list entries, allowing them to be broken across pages while the entry itself won't be. It is intended for use with, and will only properly work with, bibannotesep set to one of the modes that start a new paragraph for the annotation, to wit, par, vpar (the default), parbreak, or vparbreak. See below for the meaning of the "break" options here.
> **annotenp:** This option attempts to treat annotations separately from reference list entries in those settings of bibannotesep which don't involve starting a new paragraph. It works by setting the number of lines in an entry after which page breaking is allowed. By default entries will only break after 3 lines, the idea being that most reference list entries fit into three lines, so at that point you're likely to be inside the annotation, but you can set the entrybreak option to any integer that works for your reference apparatus.

Please note that there is one possible break point that isn't directly addressed by these options, that is, the one between the main entry and the annotation when that annotation starts a new paragraph. If you set bibannotesep to par or vpar, then LaTeX will try very hard not to break between entry and annotation, ensuring that the annotation at least starts on the same page as its entry. If you use parbreak or vparbreak, LaTeX is positively encouraged to break a page there, as is usual between paragraphs.

You can of course ignore the formatbib option and provide your own settings. Biblatex uses the `\bibsetup` command which you can renew in your preamble. You can find a nice commentary on the default values set by the package in the file biblatex.def, which you'll find in the main biblatex directory of your TeX distribution.

genallnames   This option affects the choice of which names to present in the genitive case when using the `\gentextcites` command. Please see the documentation of that command in section 5.3.2, above.

headline   Several users requested an option that turned off the automatic transformations that
(trad only)   produce sentence-style capitalization in the title fields of the 15th-edition author-date style. I have, therefore, also included it in authordate-trad. If you set this option, the word case in your title fields will not be changed in any way, that is, this doesn't automatically transform your titles into headline-style, but rather allows the .bib file to determine capitalization. It works by redefining the command `\MakeSentenceCase`, so in the unlikely event you are using the latter anywhere in your document please be aware that it will also be turned off there.

hyperall   I have received requests to allow more than one part of a citation to act as a hyperlink to the entry in the reference list (assuming you're using the hyperref package, that is). Setting this option for the whole document, for specific entry types, or for individual entries will make any names, titles, or dates present in the citation act as such a link. It acts, in other words, like a combination of hypername and hypertitle, below.

hypername This option ensures that the name component of a citation links to the entry in the reference list, again assuming that you're using hyperref. You can set it for the whole document, for specific entry types, or for individual entries. Should a citation appear that contains *only* a name, then it will always be such a link, no matter what the state of this option.

hypertitle When you use the hyperref package with the author-date styles, the in-text citations will provide a hyperlink to the full information in the list of references. Timo Thoms rightly pointed out that, generally, one only wants one piece of the citation to provide the hyperlink, usually the date part (though see the two previous options). The author-date styles will instead link the title or the shorthand if there isn't a date, but if you set this option to true globally in your preamble then all titles and shorthands will link, regardless of whether a date is also present. You can also set hypertitle by entry type or in the options field of individual entries, allowing you to provide a hyperlink in cases where the automatic mechanism gets it wrong (ency:britannica).

juniorcomma The *Manual* (6.43) states that "commas are not required around *Jr.* and *Sr.*," so by default biblatex-chicago has followed standard biblatex in using a simple space in names like "John Doe Jr." Charles Schaum has pointed out that traditional BibTeX practice was to include the comma, and since the *Manual* has no objections to this, I have provided an option which allows you to turn this behavior back on, either for the whole document or on a per-entry basis. Please note, first, that numerical suffixes (John Doe III) never take the comma. The code tests for this situation, and detects cardinal numbers well, but if you are using ordinals you may need to set this to false in the options field of some entries. Second, I have fixed a bug in older releases which always printed the "Jr." part of the name immediately after the surname, even when the surname came before the given names (as in a reference list). The package now correctly puts the "Jr." part at the end, after the given names, and in this position it always takes a comma, the presence of which is unaffected by this option.

nameaddon This option determines where and when the nameaddon field will be printed. There are three possible values, available globally, per type, and per entry:

> **all:** This is the default; if an entry has a nameaddon, it will appear in the reference list.
>
> **none:** The field will not appear in the reference list.
>
> **first:** Philipp Immel requested this as a way to provide an author's dates in the nameaddon field and only have them printed the first time that author appears in the reference list. The code tests for identical nameaddon fields in works by identical authors, so other sorts of nameaddon will be printed as usual.

Cf. nameaddonformat just below, and nameaddonsep in section 5.4.2.

nameaddon-format This option, available globally, per type, and per entry, allows you to change the format of the nameaddon field on the fly, so its value should be a field format that biblatex understands. This includes standard formats like parens, brackets or emph, and also custom formats that you provide in your preamble using \DeclareFieldFormat, in case the standard ones aren't adequate. If you don't define this option, then the usual defaults apply, that is, no formatting in online, review, and suppperiodical entries, as well as in misc entries with an entrysubtype, while square brackets surround the field in all other entry types with the exception of customc, which has its own rules and ignores this option. Cf. nameaddonsep in section 5.4.2.

natbib These are the standard biblatex options, which formerly required slightly special han-
mcite dling when you loaded the Chicago style with \usepackage{biblatex-chicago}. Both the forms natbib and natbib=true (mcite & mcite=true) should now work.

nodatebrackets When you use biblatex's enhanced date specifications to present a "circa" date ({1956~}),
noyearbrackets an uncertain date ({1956?}), or one that is both at the same time ({1956%}), the date that by default will appear in your documents will have square brackets around it. This accords with the *Manual's* instructions concerning such dates (15.44), but may in some circumstances prove syntactically awkward, or may perhaps be out of step with a specific house style. These two options, which may appear in the preamble either for the whole

document or for specific entry types, or in individual entries, allow you to control when these brackets will appear. The first controls, mainly, dates that appear in the body of an entry in the reference list, while the second controls dates in citations and at the head of entries in the reference list. Cf. table 3.

noibid   At the request of an early tester, I have included this option to allow you to turn off the `ibidem` mechanism that `biblatex-chicago-authordate` uses by default. This mechanism doesn't actually print "Ibid," but rather includes only the `postnote` information in a citation, i.e., it will print (224) instead of (Author 2000, 224). Setting this option will mean that these shortened citations no longer appear automatically. The option is settable globally, per type, or per entry, so that fine-grained control of individual citations is now possible without the use of the `\citereset` command and `biblatex citereset` option, though these are of course still available. See biblatex.pdf § 3.1.2.1.

notitle   When citing sources from antiquity (using the `classical` entrysubtype), the *Manual* (14.244–5) recommends using just the `author` in short citations if only one `title` by that author has come down to us, making the identification of the work unambiguous. I previously suggested using a command like `\citeauthor` to achieve this, but Tobias Becht suggested that a less clumsy method would be better, so I've provided the notitle option, settable by entry type and also in the `options` field of specific entries. In effect, this turns an author-title citation into an author-only one, but it won't affect author-date citations. Cf. herodotus:wilson.

ordinalgb   This option, which only affects users of the `british` language, restores the previous package defaults, printing the `day` part of a `date` specification as an ordinal number: 26th March 2017. The new package default prints 26 March 2017, which is more in keeping both with standard British usage and with the recommendations of the *Manual* (9.35). The option is available only in the preamble.

postnotepunct (experimental)   Originally designed for the notes & bibliography style, this option may in fact be more useful in the `authordate` styles. If set to `true`, it allows you to alter the punctuation that appears just before the `postnote` argument of citation commands, simplifying in particular the provision of comments within parenthetical citations. In previous releases, you either needed to include the comment after a page number, e.g., `\autocite[16; some comment]{citekey}`, or provide a separate .bib entry using the `customc` entry type, e.g., `\autocites{chicago:manual}{chicago:comment}`. With this option enabled, `\autocite[; some comment]{citekey}` will do. More generally, the `postnotepunct` option allows you to start the `postnote` field with a punctuation mark (.,;:) and have it appear as the `\postnotedelim` in place of whatever the package might otherwise automatically have chosen. Please note that this functionality relies on a very nifty macro by Philipp Lehman which I haven't extensively tested, so I'm labeling this option "experimental." Note also that the option only affects the `postnote` field of citation commands, not the `pages` field in your .bib file.

seriesabbrev   This option controls the printing, in the reference list, of the `shortseries` field in place of the `series` field in book-like entries. It is `false` by default, so as shipped `biblatex-chicago-authordate` will silently ignore such fields, but you can set it to `true` either in the preamble for the whole document or for specific entry types, or in individual entries, and it will appear in the reference list. For more details, see the documentation of `shortseries` in section 5.2, above.

shorthandfull   Kenneth Pearce has suggested that, in some fields of study, a list of shorthands providing full bibliographical information may replace the list of references itself. This option, which must be used in tandem with `cmslos=false`, prints this full information in the list of shorthands, though of course you should remember that any .bib entry not containing a `shorthand` field won't appear in such a list. Please see the documentation of the `shorthand` field in section 5.2 above for information on further options available to you for presenting and formatting the list of shorthands.

strict   This still-experimental option attempts to follow the *Manual*'s recommendations (14.36) for formatting footnotes on the page, using no rule between them and the main text unless there is a run-on note, in which case a short rule intervenes to emphasize this

continuation. I haven't tested this code very thoroughly, and it's possible that frequent use of floats might interfere with it. Let me know if it causes problems.

xrefurl Stefan Björk pointed out that when, using the longcrossref or booklongxref options, you turn on the automatic abbreviation of multiple entries in the same (e.g.) collection or mvcollection, you could entirely lose a url that might be helpful for locating a source, as the abbreviated forms in the reference list wouldn't include this information. Setting this option to true either in the preamble or in individual entries will allow the url, doi, or eprint field to appear even in these abbreviated references.

5.4.4 Style Options – Entry

These options are settable on a per-entry basis in the options field.

cmsdate Recent editions of the *Manual* have simplified the options for entries with more than one date (15.40). You can choose among them using the cmsdate entry option. It has 3 possible states relevant to this problem, alongside a fourth which I discuss below. An example should make this clearer. Let us assume that an entry presents a reprinted edition of a work by Smith, first published in 1926 (the origdate) and reprinted in 1985 (the date):

off: This is the default. The citation will look like (Smith 1985).

both: The citation will look like (Smith [1926] 1985).

on: The citation will look like (Smith 1926). NB: The *Manual* no longer includes this among the approved options. If you want to present the origdate at the head of an entry, then generally speaking you should probably use cmsdate=both. I have retained the option because in some cases it is still useful. The old options new and old work like both.

As I explained in detail above in section 5.2, s.v. "**date**," because biblatex's sorting algorithms and automatic creation of the extradate field refer by default to the date before the origdate when both are present, there may be situations when you need to have the *earlier* year in the date field, and the later one in origdate, e.g., if you have another reprinted work by the same author originally printed in the same year. Biblatex-chicago-authordate will automatically detect this switch, and given the same reprinted work as above, the results will be as follows:

off: This is the default. The citation will look like (Smith 1926a). This style is no longer recommended by the *Manual.*

both: The citation will look like (Smith [1926a] 1985). The old options old and new are synonyms for this.

on: The citation will look like (Smith 1926a). As noted above, this style is no longer recommended by the *Manual.*

switchdates If, for any reason, simply switching the date and the origdate isn't possible in a given entry, then you can put switchdates in the options field to achieve the same result. Also, you can cmsdate use the preamble version of cmsdate to change the default order of \DeclareLabeldate, *in preamble* generally making this date-switching in your .bib file unnecessary. Please take a look at the full documentation of the **date** field to which I referred just above, at the preamble cmsdate documentation in section 5.4.3, and also at cms-dates-sample.pdf and dates-test.bib for examples of how all this works.

Bertold Schweitzer has brought to my attention certain difficult corner cases involving cross-referenced works with more than one date. In order to facilitate the accurate presentation of such sources, I made a slight change to the way cmsdate=on and cmsdate=both work. If, and only if, a work has only one date, and there is no switchdates in the options field, then cmsdate=on and cmsdate=both will both result in the suppression of the extradate field in that entry. Obviously, if the same options are set in the preamble, this behavior is turned off, so that single-date entries will still work properly without manual intervention.

Recent editions of the *Manual* specify that it is "usually sufficient to cite newspaper and magazine articles entirely within the text" (15.49). This will apply mainly to article and review entries with entrysubtype magazine, and involves a parenthetical citation giving the journaltitle and then the full date, not just the year, with any other relevant identifying information incorporated into running text. (Cf. 14.198.) In order to facilitate this, I cmsdate=full have added a further switch to the cmsdate option — full — which *only* affects the presentation of citations, and causes the printing of the full date specification there. You can use the standard biblatex skipbib option to keep such entries from appearing in the list of references, and you may, if your .bib entry is a complete one, also need useauthor=false in order to ensure that the journaltitle appears in the citations rather than the author.

5.5 General Usage Hints

5.5.1 Loading the Styles

With the addition of the authordate-trad style to the package, there are now three keys for choosing which style to load, notes, authordate, and authordate-trad, one of which you put in the options to the \usepackage command. With early versions of biblatex-chicago, the standard way of loading the package was via a call to biblatex, e.g.:

```
\usepackage[style=chicago-authordate,strict,backend=biber,%
  babel=other,bibencoding=inputenc]{biblatex}
```

Now, the default way to load the style, and one that will in the vast majority of standard cases produce the same results as the old invocation, will look like this:

```
\usepackage[authordate,strict,backend=biber,autolang=other,%
  bibencoding=inputenc]{biblatex-chicago}
```

If you read through biblatex-chicago.sty, you'll see that it sets a number of biblatex options aimed at following the Chicago specification, as well as setting a few formatting variables intended as reasonable defaults (see section 5.4.1, above). Some parts of this specification, however, are plainly more "suggested" than "required," and indeed many publishers, while adopting the main skeleton of the Chicago style in citations, nonetheless maintain their own house styles to which the defaults I have provided do not conform.

If you only need to change one or two parameters, this can easily be done by putting different options in the call to biblatex-chicago or redefining other formatting variables in the preamble, thereby overriding the package defaults. If, however, you wish more substantially to alter the output of the package, perhaps to use it as a base for constructing another style altogether, then you may want to revert to the old style of invocation above. You'll lose all the definitions in biblatex-chicago.sty, including those to which I've already alluded and also the code that sets the note number in-line rather than superscript in endnotes or footnotes, the URL line-breaking code, and the Chicago-specific number- and date-range compression code. You'll need to load the required packages xstring and nameref yourself, as biblatex doesn't do it for you. Also in this file is the code that calls all of the package's localization files, which means that you'll lose all the Chicago-specific bibstrings I've defined unless you provide, in your preamble, a \DeclareLanguageMapping command, or several, adapted for your setup, on which see section 7 below and also §§ 4.9.1 and 4.11.8 in biblatex.pdf.

What you *will not* lose is the ability to call the package options annotation, strict, cmslos=false and noibid (section 5.4.3, above), in case these continue to be useful to you when constructing your own modifications. There's very little code, therefore, actually in biblatex-chicago.sty, but I hope that even this minimal separation will make the package somewhat more adaptable. Any suggestions on this score are, of course, welcome.

5.5.2 Other Hints

Starting with biblatex version 1.5, in order to adhere to the author-date specification you will need to use Biber to process your .bib files, as BIBTEX (and its more recent variants) will no longer provide all the required features. This document assumes that you are

using Biber; if you wish to continue using BibTeX then you need biblatex version 1.4c and biblatex-chicago 0.9.7a.

If your .bib file contains a large number of entries with more than three authors, then you may run into some limitations of the biblatex-chicago code. The default settings are `maxnames=3,minnames=1` in citations and `maxbibnames=10,minbibnames=7` in the list of references. In practice, this means that an entry like hlatky:hrt, with 5 authors, will present all of them in the list of references but will truncate to one in citations, like so: (Hlatky et al. 2002). For the vast majority of circumstances, these settings are exactly right for the Chicago author-date specification. However, if "a reference list includes another work of the same date that would also be abbreviated as ['Hlatky et al.'] but whose coauthors are different persons or listed in a different order, the text citations must distinguish between them" (15.29). The (Biber-only) biblatex option `uniquelist`, set for you in biblatex-chicago.sty, will automatically handle many of these situations for you, but it is as well to understand that it does so by temporarily suspending the limits, listed above, on how many names to print in a citation. Without `uniquelist`, biblatex would present such a work as, e.g., (Hlatky et al. 2002b), while hlatky:hrt would be (Hlatky et al. 2002a). This does distinguish between them, but inaccurately, as it suggests that the two different author lists are exactly the same. With `uniquelist`, the two citations might look like (Hlatky, Boothroyd et al. 2002) and (Hlatky, Smith et al. 2002), which is what the specification requires.

If, however, the distinguishing name occurs further down the author list — in fourth or fifth position in our examples — then the default settings would produce citations with all 4 or 5 names printed, which can become awkwardly long. In such a situation, you can provide shortauthor fields that look like this: {{Hlatky et al., \mkbibquote{Quality of Life,}}} and {{Hlatky et al., \mkbibquote{Depressive Symptoms,}}}, using a shortened title to distinguish the references. This would produce (Hlatky et al., "Quality of Life," 2002) and (Hlatky et al., "Depressive Symptoms," 2002), again as the spec requires. There is, unfortunately, no simpler way that I know of to deal with this situation.

One useful rule, when you are having difficulty creating a .bib entry, is to ask yourself whether all the information you are providing is strictly necessary. The Chicago specification is a very full one, but the *Manual* is actually, in many circumstances, fairly relaxed about how much of the data from a work's title page you need to fit into a reference. Authors of introductions and afterwords, multiple publishers in different countries, the real names of authors more commonly known under pseudonyms, all of these are candidates for exclusion if you aren't making specific reference to them, and if you judge that their inclusion won't be of particular interest to your readers. Of course, any data that may be of such interest, and especially any needed to identify and track down a reference, has to be present, but sometimes it pays to step back and reevaluate how much information you're providing. I've tried to make biblatex-chicago robust enough to handle the most complex, data-rich citations, but there may be instances where you can save yourself some typing by keeping it simple.

Scot Becker has pointed out to me that the inverse problem not only exists but may well become increasingly common, to wit, .bib database entries generated by bibliographic managers which helpfully provide as much information as is available, including fields that users may well wish not to have printed (ISBN, URL, DOI, pagetotal, inter alia). The standard biblatex styles contain a series of options, detailed in biblatex.pdf § 3.1.2.2, for controlling the printing of some of these fields, and I have implemented others that are relevant to biblatex-chicago, along with a couple that Scot requested and that may be of more general usefulness. There is also a general option to excise with one command all the fields under consideration – please see section 5.4.2 above.

6 The Jurisdiction, Legislation, and Legal Entry Types

I have received numerous requests over the years to include some means of referring to legal and public documents which, broadly speaking, don't fit easily into any of the standard biblatex entry types. The *Manual* (14.269–305) recommends using the *Bluebook* as a guide for formatting such references, while also suggesting certain modifications to

this formatting to bring it more into line with Chicago's usual practices. Biblatex-chicago now offers three entry types — **jurisdiction**, **legal**, and **legislation** — which allow you to present at least a substantial subset of what the *Bluebook* offers. As the rules for your .bib entries are the same in the notes & bibliography style and in the author-date styles, and as these rules mainly come from a source outside the *Manual,* and additionally as these rules apparently require even the author-date styles to use a system of foot- or endnotes (15.58), I have documented these types in a section of their own, applicable to all the Chicago styles. (Some few changes needed when using the author-date styles, mainly to do with citation commands, will be outlined at the end.) You can also consult the example files legal-test.bib and cms-legal-sample.pdf to see how you might construct your database entries.

6.1 Types, Subtypes, and Fields

Anyone who has used the *Bluebook* will realize that it is hopeless to attempt to fit its labyrinthine complexities into three entry types, but with the addition of numerous entrysubtypes and some parsing by Biber under the hood, I hope to have covered the main sorts of material discussed by the *Manual.* As a first approximation, all three types begin from a structure analogous to the standard biblatex article type, with a number of subtle differences that I have attempted to make consistent across the three. Standard practice is to present the references *only* in notes, and not in a bibliography, so by default biblatex-chicago excludes these types from the latter, though you can control this using an option (see below).

jurisdiction   This type is for presenting legal cases and court decisions. A typical entry will contain the following fields:

title: The case name as seen in the first, long note.

shorttitle: The case name for subsequent, short notes, ordinarily either the plaintiff or the nongovernmental party.

journaltitle and/or shortjournal: The reporter for the case, *always* presented in a standard abbreviated form available in the *Bluebook.* You can place the abbreviation either in the journaltitle or in the shortjournal field. If you wish to present your readers with a list of abbreviations with their expansions, then the expansion goes in journaltitle and the abbreviation in shortjournal. (Cf. cms-legal-sample.pdf to see how this might look.)

pages *or* issue: When using a standard official reporter, this will contain the opening page of the decision in that reporter, while any postnote field will contain the specific page on which a particular citation appears (a "pincite"). When citing a commercial electronic database, on the other hand, you should give, instead of a pages field, the identifying number of the case using the issue field. Biblatex-chicago uses the presence of the issue field to provide the slightly different formatting required for citations from databases like Westlaw or LexisNexis. When an issue field is present, then both the pages field and the postnote field can provide a pincite. (Cf. federal:case and database:case.)

series: If you are citing an official reporter, then it may have a series number, which will be printed immediately after the name of the reporter.

volume: The volume number of the reporter. It will often be the same as the year when using a commercial electronic database, but you still need to provide it separately.

number: The docket number of the case, generally required when the reporter is a commercial database.

date: The date of the decision.

location: The abbreviated name of the court, if it isn't clear from the reporter cited. It will be associated with the date in American cases, but not in Canadian or UK cases (see below). Being a list field, it can contain more than one item, in case you need a separate set of parentheses to identify a jurisdiction as well as a court name in Canadian or UK cases (cf. uk:case:square).

These are, so to speak, the basic elements of a jurisdiction citation, which may, depending on specific circumstances, require supplementation by the following:

entrysubtype{square} or {round}: The *Manual* includes examples for citing cases in the United Kingdom and in Canada, and the square entrysubtype identifies the reporter either as Canadian or as a UK reporter for which the year is essential to locating the case, that is, when "there is either no volume number or the volumes for each year are numbered anew, not cumulatively" (14.298). The round subtype, by contrast, identifies a UK reporter where the volumes are numbered cumulatively, making the year inessential. (The names refer to the shapes of the brackets placed around the year in each case. Cf. canada:case, uk:case:round, and uk:case:square.)

origlocation or origpublisher: If you need to cite more than one reporter for a given case, then there are two possibilities. Either the second (and subsequent) reporter(s) use(s) the same pagination as the first (origpublisher) or the reporters use different pagination (origlocation). Since both are list fields, you can in theory provide several reporters, but please note that these fields are currently only provided for citations of American cases. (Cf. state:case:2reps.)

related: It may be necessary sometimes to indicate further action by another, higher court, such as the US Supreme Court's grant or denial of *certiorari*. The usual related mechanism is useful in such situations, particularly with a tailored relatedstring field. (Cf. federal:lower:related.)

legislation   This is the most complicated of the new entry types, with several entrysubtypes and a number of tricky corners, particularly with regard to the provision of subsequent short notes after the first full citation. It is intended to cope with constitutions and with legislative and executive documents of all kinds, with the exception of treaties, for which you can use the legal type, below. In effect, the type tries to cover federal, state, and municipal laws and ordinances, statutes, bills, resolutions, reports, debates, hearings, presidential and congressional documents, and constitutions, none of which it does with particular elegance, so consider it a work in progress. Many of the fields have close analogues in the jurisdiction type, so at least there is some bare minimum of consistency when dealing with public and legal material.

author: Some kinds of material, usually reports, may have an author, often an organizational one. (Cf. congress:report and uk:command.)

title: Reports, bills, hearings and the like frequently have a title which, please note, quite frequently will not turn up in short notes, depending on which other fields are present.

titleaddon: This field is considerably more important in legislation entries than the shorttitle field, mainly because it will turn up in many short notes where the title will not. It will frequently contain specifying information on legislative material, and will therefore often allow short notes to differentiate citations of sources that might have the same title but differ in other respects. (Cf. congress:publiclaw, congress:bill, and congress:report.)

number: Gives an identifying number to a title or a titleaddon, with \bibstring{number} prefixed to it. It too can appear in short notes.

note: This gives a section or other specifying information related to a titleaddon and number. (Cf. congress:publiclaw.)

journaltitle and/or shortjournal: There is usually a standard place for publishing various sorts of legislative material, and as in jurisdiction entries it is *always* presented in a standard abbreviated form available in the *Bluebook*. You can place the abbreviation either in the journaltitle or in the shortjournal field. If you wish to present your readers with a list of abbreviations with their expansions, then the expansion goes in journaltitle and the abbreviation in shortjournal. Like titleaddon, this field will often appear in short notes. (Cf. cms-legal-sample.pdf to see how this might look.)

volume: The volume number of the shortjournal. It can be a cardinal or an ordinal, depending on the shortjournal. (Cf. congress:publiclaw and congress:bill.)

series: In citations of American material, this will usually contain session information pertaining to a legislative publication. Elsewhere it will often contain just be a plain number, not unlike in jurisdiction entries. (Cf. congress:debate:globe, state:statute:okla, and uk:hansard.)

issue: This field can provide an identifying number in some circumstances, particularly when you don't want it prefixed by any bibstring — cf. uk:command.

pages: Somewhat similar to its use in jurisdiction entries, this will usually contain the opening page, or sometimes the section number, of the material in the shortjournal, while any postnote field will contain the specific page on which a particular citation appears (a "pincite").

part or chapter: Some sources use part or chapter numbers instead of pages or sections. (Cf. canada:statute and uk:statute.)

date: The date of publication of the material, usually just a year, though sometimes a full date, e.g., see executive:proclamation.

location: If it is not clear from the title or the shortjournal, this field can specify, in abbreviated form, the US state where the legislative material originates. It will be associated with the date in long notes, but will appear elsewhere in short notes. (Cf. state:statute:okla.)

usera: This specifies a particular edition, possibly from a commercial electronic database, of a legislative publication. It will be associated with the date in long notes but won't appear in short ones. (Cf. congress:debate:new and state:statute:ky.)

addendum: You can use this field to specify the speaker at hearings or in debates, the Canadian or British jurisdiction of some laws if not otherwise clear from the citation, or possibly simply additional information about a source. (Cf. canada:statute, congress:debate:new, congress:hearing, state:statute:ky, and uk:statute.)

entrysubtype: The sheer variety of sources included under the legislation type, and the specialized rules for presenting them, have necessitated the introduction of a substantial network of entrysubtypes:

canada: Identifies Canadian statutes (canada:statute).

constitution: For constitutions, be they federal, state, or local (constitution:arkansas and constitution:federal). Biber will automatically detect if the title contains the string Const and provide the entrysubtype for you, but in other cases you'll have to provide it yourself.

hansard: Identifies UK parliamentary debates as published in *Hansard*. (Cf. uk:hansard.)

hearing: For congressional hearings (congress:hearing).

uk: Identifies UK statutes and command papers. (Cf. uk:command, uk:statute, and uk:statute:regnal.)

un: For UN documents (un:resolution).

A glance through the legal-test.bib file should help enormously when you're trying to work out how to present a particular source, and all suggestions for pruning the foliage will be welcome.

legal This type is intended as a catch-all for miscellaneous public documents not included in the previous two types, but for the moment the only sort of material for which it is required is international treaties (14.290; treaty). The usual fields for such material include:

title: The treaty name as seen in the first, long note.

shorttitle: The treaty name for subsequent, short notes. You can also use the shorthand field in such entries.

titleaddon: This contains the names of the countries involved in the treaty, in abbreviated form.

journaltitle and/or shortjournal: The standard publication containing the treaty, *always* presented in an abbreviated form available in the *Bluebook*. You can place the abbreviation either in the journaltitle or in the shortjournal field. If you wish to present your readers with a list of abbreviations with their expansions, then the expansion goes in journaltitle and the abbreviation in shortjournal. (Cf. cms-legal-sample.pdf to see how this might look.)

issue: This will contain the identifying number provided by the shortjournal.

volume: The volume number of the shortjournal.

date: The exact date of signing, as the year of publication can differ from it.

pages: This, or a postnote field, can contain a specific page reference ("pincite").

6.2 Citation Commands

The *Bluebook* style mandates footnotes without a bibliography, so it should be simple to include such references in the Chicago notes & bibliography style, which uses foot- or endnotes as standard. The usual citation commands should work as you expect, though I wouldn't recommend the \textcite commands, as they will produce surprising and unsatisfactory results. For users of the author-date styles, however, the *Bluebook* more or less requires you to adopt a separate set of notes in addition to the standard author-year citations, which means that for these three legal entry types you'll have to remember to use new citation commands that I've provided: \fullcite, \footfullcite, and \parenfullcite. The first prints the reference, the second does so in a footnote, and the third does

\runcite so inside parentheses. For both styles, there is a new command that you should use if you are citing a jurisdiction entry in the running flow of text rather than as a stand-alone citation, whether that text is in a note or in the main body (14.276). This requires a different presentation of the title field, and using \runcite alerts biblatex-chicago to this circumstance.

6.3 Options

Several new options allow you to control the presentation of legal notes in your document. The default settings are indicated in the margins.

legalnotes=true This option prevents the printing of legal citations in a bibliography or reference list, as the *Bluebook* recommends. You can change this to false in the preamble of your document, but you should be aware that the reference printed in the bibliography will be a clone of the long-note form, as the *Bluebook* doesn't provide an alternative version.

noneshort= This option controls the availability of the short form of the note, intended for use in sub-
false sequent citations of entries already presented in full notes. By default, biblatex-chicago attempts to provide *Bluebook*-authorized short versions of citations in this situation, and some of the many entrysubtypes are at least partially designed to cope with the complexities of the specification. The *Manual*, for its part, suggests that "works that cite only a few legal documents may be better off using the full form for each citation" (14.275). In the author-date styles, you can set this option to true globally, per type, or in individual .bib entries to accomplish this. Assuming you've only used the \fullcite commands for the *Bluebook* entry types, the option will only apply to such entries. In the notes & bibliography style the global option would apply to all entry types, but you can also set this to true on a type-by-type basis or in individual entries. It is designed mainly for use with *Bluebook* entries, but it might perhaps be useful elsewhere. Please be aware that, even with this option on, the *ibidem* mechanism remains in operation for repeated citations, and also that the option may give surprising results in the presence of shorthand fields and/or the shorthandfirst and short options.

short=false I have ported this option, already present in the notes & bibliography style, to the author-date styles to allow users to present short notes from the very first citation. I'm not certain what the use case might be for this, as it's intended for saving space in documents where short notes can point to references in a full bibliography. Still, if for any reason you need this you can set the option to true globally, by entry type, or for specific entries.

supranotes= This is a *Bluebook*-specific option, and it produces, for some entry types and subtypes, a
true back reference to the first, long note at the end of subsequent, short citations. It takes the form "*supra* note #," and is available in all Chicago styles, though you'll only see it in certain sorts of citation, automatically controlled by biblatex-chicago in accordance with the *Bluebook* specification. If you prefer not to see such back references, you can set the option to false either in the preamble or in individual .bib entries.

7 Internationalization

Several users have requested that, in line with analogous provisions in other "American" biblatex styles (e.g., biblatex-apa and biblatex-mla), I include facilities for producing a Chicago-like style in other languages. I have supplied three lbx files, cms-german.lbx, its clone cms-ngerman.lbx, and cms-french.lbx, in at least partial fulfillment of this request.

For this release Marçal Orteu Punsola has provided cms-spanish.lbx, thereby adding to the generous contributions of Patrick Danilevici (cms-romanian.lbx), Wouter Lancee (cms-dutch.lbx), Gustavo Barros (cms-brazilian.lbx), Stefan Björk (cms-swedish.lbx), Antti-Juhani Kaijahano (cms-finnish.lbx), Baldur Kristinsson (cms-icelandic.lbx), and Håkon Malmedal (cms-norsk.lbx, cms-nynorsk.lbx, and cms-norwegian.lbx). I include cms-british.lbx in order to simplify and to improve the package's handling of non-American typographical conventions in English. This means that all — or at least most — of the Chicago-specific bibstrings are now available for documents and reference apparatuses written in these languages, with, as I intend, more languages to follow, limited mainly by my finite time and even-more-finite competence. (If you would like to provide bibstrings for a language in which you want to work, or indeed correct deficiencies in the lbx files contained in the package, please contact me.)

babel Using these facilities is fairly simple. By default, and this functionality remains the same as it was in the previous release of biblatex-chicago, calls to \DeclareLanguageMapping in biblatex-chicago.sty will automatically load the American strings, and also biblatex's American-style punctuation tracking, when you:

1. Load babel with american as the main text language.

2. Load babel with english as the main text language.

    or

3. Do not load babel at all.

(This last is a change from the biblatex defaults — cp. § 3.12.2 in biblatex.pdf — but it seems to me reasonable, in an American citation style, to expect this arrangement to work well for the majority of users.)

If, for whatever reason, you wanted to use biblatex-chicago but retain British typographical conventions — punctuation outside of quotation marks, outer quotes single rather than double, etc. — then you no longer need to follow the complicated rules outlined in previous releases of biblatex-chicago. Instead, simply load babel with the british option.

If you want to use Brazilian Portuguese, Dutch, Finnish, French, German, Icelandic, Norwegian, Romanian, Spanish, or Swedish strings in the reference apparatus, then you can load babel with brazilian, dutch, finnish, french, german, icelandic, ngerman, norsk, nynorsk, romanian, spanish, or swedish as the main document language. You no longer need any calls to \DeclareLanguageMapping in your document preamble, since biblatex-chicago.sty automatically provides these if you load the package in the standard way.

You can also define which bibstrings to use on an entry-by-entry basis by using the hyphenation field in your bib file, but you will have to make sure that all the languages you want to use in this way are included in the call to load babel in your preamble, even if not as the main text language. The \DeclareLanguageMapping calls in biblatex-chicago.sty should do the rest, assuming you've loaded the package that way.

Three other hints may be in order here. Please note, first, that I haven't altered the standard punctuation procedures used in any of the other available languages, so commas and full stops will appear outside of quotation marks, and those quotation marks themselves will be language-specific. If, for whatever reason, you wish to follow the Chicago specification and move punctuation inside quotation marks, then you'll need a declaration of this sort in your preamble:

```
\DefineBibliographyExtras{german}{%
  \DeclareQuotePunctuation{.,}}
```

Second, depending on the nature of your bibliography database, it will only rarely be possible to process the same bib file in different languages and obtain completely satisfactory results. Fields like note and addendum will often contain language-specific information that won't be translated when you switch languages, so manual intervention will be necessary. If you suspect you may have a need to use the same bib file in different languages,

you can minimize the amount of manual intervention required by using the bibstrings defined either by biblatex or by biblatex-chicago. Here, a quick read through notes-test.bib and/or dates-test.bib should give you an idea of what is available for this purpose — see esp. the strings by, nodate, newseries, number, numbers, oldseries, pseudonym, reviewof, revisededition, and volume, and also section 4.3.1 above, esp. s.v. "\partedit."

Finally, the French and German bibstrings I have provided may well break with established bibliographical traditions in those languages, but my main concern when choosing them was to remain as close as possible to the quirks of the Chicago specification. I have relied on the judgment of the creators of the Brazilian, Dutch, Finnish, Icelandic, Norwegian, Romanian, Spanish, and Swedish localizations in those instances, as well as attempting to maintain harmony where necessary with the standard biblatex translations. If you have strong objections to any of the strings, or indeed to any of my formatting decisions, please let me know.

7.1 The cmsnameparts Option

Biblatex provides a mechanism for localizing name presentation in citations and in bibliographies, so for example making it possible to present Russian patronymics or Chinese names (both in the original script and in transliteration) in ways that the standard macros can't provide, focusing as those macros do on the usual Western European approaches. The files 93-nameparts.tex and 93-nameparts.dbx outline and demonstrate the changes required, providing code for Russian, Ethiopian, and CJK names. The *Manual* (11.90) too provides some examples, admitting the necessity of such treatments in some contexts. At the request of Philipp Immel, I have provided the cmsnameparts option in all styles to facilitate access to this mechanism for biblatex-chicago users. In practice what I have done is the following:

1. Include 93-nameparts.dbx in the package under the name cms.dbx. This extends the default data model, adding new kinds of name part to it.

2. Copy the code from 93-nameparts.tex into the .cbx files, along with some shims that allow the addition of new name templates and also allow any of the templates to serve as the default template, that is, the template to be used when a .bib entry doesn't specify one.

3. Define the cmsnameparts option in such a way that none of this code will be loaded unless you call the option, which meant that I had to define it in biblatex-chicago.sty instead of in the individual styles themselves. If you don't load the style in the standard way with a call to biblatex-chicago then a little more work will be required.

cmsnameparts The basic usage is straightforward. If you just put cmsnameparts among the options when loading biblatex-chicago and add no additional options to your .bib entries, nothing visible will change. Any entry that contains one of biblatex's nametemplate options will then be able to format its names according to the Russian, Ethiopian, or CJK specifications. You can see this in action in the entry hua:cms in the biblatex-chicago documentation (cms-notes-sample.pdf and cms-dates-sample.pdf). It uses nametemplates=cjk in its options field, which is the simplest approach, though more fine-grained approaches are possible with the biblatex options sortingnamekeytemplate and uniquenametemplatename (biblatex.pdf § 3.8.10 and appendix E; cf. also 93-nameparts.tex). Please note that when you use a non-standard name part in a name list, you'll need to identify even those standard parts that are present, e.g., family=Hua, given=Linfu.

cmsnameparts= It's possible that you might have a .bib database for a specific project that contains a
<nametemplate> majority of entries requiring a change of name template. In this case you can change the default template by giving its name to the cmsnameparts option. Any entry not having a nametemplate option will then use the default, which may save you some typing. If an entry requires the biblatex standard default template, its name, following the example of 93-nameparts.tex, is western.

It's also perfectly possible that none of the available templates will suit your needs. The simplest course, at least for you, is to send me a feature request and a fairly thorough description of the specification, and I'll try to add it to the list of available templates. If you prefer to code it yourself while still using the cmsnameparts mechanism, you'll need to look over the code in chicago-notes.cbx or chicago-dates-common.cbx and provide several macros according to a specific naming scheme. You'll need at least:

```
\DeclareSortingNamekeyTemplate[<templatename>]{<specification>}
\DeclareUniquenameTemplate[<templatename>]{<specification>}
\newbibmacro*{name:<templatename>}[<no. of nameparts as digit>]{<code>}
\csdef{cmssort:<templatename>}{\usebibmacro{name:<templatename>}...}
\csdef{cmslabel:<templatename>}{...\usebibmacro{name:<templatename>}...}
```

If your name specification reverses the order of the first name in the bibliography or reference list (as in the western style), then you'll also need:

```
\newbibmacro*{name:<templatename>-rev}[<no. of nameparts as digit>]{<code>}
\csdef{cmssort:<templatename>-rev}{\usebibmacro{name:<templatename>-rev}...}
```

If you want to be able to use the your new template name as the default template, you'll need to wrap the first two declarations like so:

```
\cms@template@wrapper{\DeclareSortingNamekeyTemplate}[<templatename>]{<spec>}
\cms@template@wrapper{\DeclareUniquenameTemplate}[<templatename>]{<spec>}
```

If your specification requires new name parts, you can copy cms.dbx to your working directory and add them to \DeclareDatamodelConstant there. Similarly, if you already load your own datamodel (.dbx) file using the datamodel option, then if you want to use cmsnameparts, which calls that option internally, you should add the code from your own .dbx file to cms.dbx and clear the datamodel option from your preamble. (That option can only load one file.)

Finally, a few notes if you use biblatex-chicago without loading biblatex-chicago.sty. You can easily load cms.dbx using biblatex's datamodel option, or else just include the declaration there in your own .dbx file. In order to load the code in the .cbx files, however, you'll need to define \cms@ldt@cmsnameparts in your preamble before loading biblatex. A line like:

```
\gdef\cms@ldt@cmsnameparts{western}
```

should do, or use another template name if you want something else as the default.

No matter how you load the code, I recommend at least a quick read of 93-nameparts.tex just to get an idea of how to construct your .bib entries. If your entries are entirely devoid of western script, you may need a \DeclareSortTranslit in your preamble to tell biblatex how to sort such entries when mixed with other scripts. An example of its use for Russian entries can be found in that file. More generally, the nameparts feature is new to biblatex-chicago, so please let me know if something doesn't work as you expect.

8   One .bib Database, Two Chicago Styles

I have, when designing this package, attempted to keep at least half an eye on the possibility that users might want to re-use a .bib database in documents using the two different Chicago styles. The extensive unification of the two styles in recent editions of the *Manual* has simplified things, and though I have no idea whether this will even be a common concern, I still thought I might gather in this section the issues that a hypothetical user might face. The two possible conversion vectors are by no means symmetrical, so I provide two lists, items within the lists appearing in no particular order. These may well be incomplete, so any additions are welcome.

8.1 Notes -> Author-Date

This is, I believe, the simpler conversion, as most well-constructed .bib entries for the notes & bibliography style will nearly "just work" in author-date, but here are a few caveats nonetheless:

1. **NB:** Unless you are using authordate-trad, the formatting of titles in the two styles is now the same, which means you would no longer need to worry about extra curly brackets and their effects on capitalization. If you are using authordate-trad, please see the caveats in the documentation of the title field in section 5.2, above.

2. You may need to reevaluate your use of shorthands, given that by default the author-date styles use them in place of authors rather than in place of the whole citation. The preamble option cmslos=false may help, but this may leave your document out-of-spec.

3. The potential problem with multiple author lists containing more than three names doesn't arise in the notes & bibliography style, so the shortauthor fields in such entries may need alteration according to the instructions in section 5.5.2 above.

4. Date presentation is relatively simple in notes & bibliography, so you'll need to contemplate the cmsdate options from sections 5.4.3 and 5.4.4 when doing the conversion to author-date.

8.2 Author-Date -> Notes

It is my impression that an author-date .bib database is somewhat easier to construct in the first instance, but subsequently converting it to notes & bibliography is a little more onerous. Here are some of the things you may need to address:

1. If you've decided against using the \partedit macro and friends from section 5.3.1 above, commands not strictly necessary for author-date, you'll need to insert them now.

2. In general, you need to be more careful in notes & bibliography about capitalization issues. Fields which only appear once in author-date — in the list of references — may appear in both long notes and in the bibliography, in different syntactic contexts, so a quick perusal of the documentation of the \autocap macro in section 5.3.1 above may help.

3. You also need to be more careful about the use of abbreviations, e.g., in journal names, where the author-date style is more liberal in their use than the notes & bibliography style. (Cf. 14.170.) The bibstrings mechanism and package options sort much of this out automatically, but not all.

4. The shorttitle field is used extensively in notes & bibliography to keep short notes short, so you may find that you need to add a fair number of these to an author-date database. In general this field is ignored by the latter style, so this, too, will be a one-time conversion.

5. You may need to add letter entries if you are citing just one letter from a published collection. See section 4.1, s.v. "letter," above.

6. The default shorthand presentation differs from one style to the other. You may need to reconsider how you use this field when making the conversion.

7. As I explained above in section 4.2, s.v. "date," I have included compatibility code in biblatex-chicago-notes for the cmsdate (silently ignored) and switchdates options, along with the automatic mechanism for reversing date and origdate. This means that you can, in theory, leave all of this alone in your .bib file when making the conversion, though I'm retaining the right to revise this if the code in question demonstrably interferes with the functioning of the notes & bibliography style.

9 Interaction with Other Packages

endnotes For users of the endnotes package — or of pagenote — biblatex offers extensive compatibility options. Please read the documentation of the notetype option in biblatex.pdf § 3.1.2.1. If you are using the noteref option with the notes & bibliography style and endnotes, please read the documentation in section 4.4.4 for your options, which include the cmsendnotes.sty package.

memoir Another problem I have found occurs because the memoir class provides its own commands for the formatting of foot- and end-note marks. By default, biblatex-chicago uses superscript numbers in the text, and in-line numbers in foot- or end-notes, but I have turned this off when the memoir class is loaded, reasoning that users of that package may well have their own ideas about such formatting.

ragged2e The footnote mark code I've just mentioned also causes problems for the ragged2e package, but in this case a simple workaround is to load biblatex *after* you've loaded ragged2e in your document preamble. The URL line-breaking code activated by cmsbreakurl doesn't play well with ragged2e, and so far I've not found a workaround.

10 TODO & Known Bugs

This release implements the 17th edition of the *Chicago Manual of Style.* It contains a version of the author-date style (authordate-trad) with traditional title formatting, alongside the authordate code which unifies the treatment of titles between itself and the notes & bibliography style. I strongly encourage users to migrate to one of the styles implementing the most recent specification, as I am focusing all of my development and testing time there.

Regardless of which edition you are considering, there remain things I haven't implemented. The solution in brown:bremer to multi-part journal articles obviously isn't optimal, and I should investigate a way of making it simpler. If the kludge presented there doesn't appeal, you can always, for the time being, refer separately to the various parts. If you have other issues with particular sorts of citation, I'm of course happy to take them on board. The *Manual* covers an enormous range of materials, but it seems to me that the available entry types could be pressed into service to address the vast majority of them. If this optimism proves misguided, please let me know.

Kenneth L. Pearce has reported a bug that appears when using multiple citation commands inside the annotation field of annotated bibliographies. I have tried to remedy this with the \citeincite and \fullciteincite commands, which print short and long citations, respectively. The first is intended for use inside a small selection of fields, while the long version should only be used in the annotation field. See sections 4.3.2 and 5.3.2, above, for details. If you still encounter problems with these commands then it may be worth setting the biblatex option casechanger=latex2e in the preamble to see if that helps, though in my testing the two different case-changing backends are now equivalent in this regard.

Roger Hart, Pierric Sans, and a number of other users have reported a bug in the formatting of title fields. This, as far as I can tell, has to do with the interaction between \MakeSentenceCase and certain characters at the start of the title, particularly Unicode ones. If you are using authordate-trad, it may help for the moment to put an empty set of curly braces {} at the start of the field, but I shall look into this further.

Patrick Danilevici's Romanian localization (cms-romanian.lbx) required some fairly extensive changes to the basic package code, so it's possible that I could improve the package's handling of other languages in a similar way.

This release fixes the formatting errors of which I am aware. There remain the larger issues I've discussed throughout this documentation, which mainly represent my inability to make all of biblatex-chicago's formatting functions transparent for the user, but thankfully biblatex's superb punctuation-tracking code preemptively fixed a great many

small errors, some of which I hadn't even noticed before I began testing that functionality. That there are other micro-bugs seems certain — if you report them I'll do my best to fix them.

11   Revision History

2.3a: Released November 17, 2022

This release backports a bug-fix from the most recent LaTeX3 programming layer. If you are using biblatex 3.18b and an older L3 layer then the processing of your documents could fail and fall into an infinite loop, leaving behind an inscrutable message in your log file. This update should prevent that, and also fixes a few other minor glitches.

2.3: Released July 2, 2022

All styles require the current biblatex (3.18) and biber (2.18).

- Since the release of biblatex 3.17, the full array of ISO8601-2 year divisions (seasons, quarters, quadrimestrals, semestrals, etc.) is available to its users, and biblatex-chicago has followed suit in all styles. The *Manual* itself, as far as I know, mentions only the seasons, but if you use a month specification higher than 24 in a date field you can access these new divisions. Cf. Table 3.

- Since biblatex provides a mechanism for localizing name presentation in the reference apparatus, Philipp Immel suggested I add an option to facilitate access to it for users of biblatex-chicago. The cmsnameparts option is the result, and is available in all styles. Please consult section 7.1 above for all the details.

- Philipp also pointed out that the name parts code requires the standard biblatex option uniquename to be turned on in the notes & bibliography style. (It's now set to minfull.) This fixes an ancient bug, as Chicago has always required authors with the same surname to be distinguished by initials or full given names in short notes (14.32).

- Patrick Danilevici pointed out that, in languages that don't use the Oxford comma, it was generally wrong to have a comma after the first, reversed name in an author list consisting of only two names. I have removed it in such languages, but if you still want the old behavior you can use \DeclareDelimFormat{revsdnamedelim} to set it to \addcomma.

- After a request by Ryo Furue, I've added the onlynd switch to the doi package option in all styles in order to suppress both the url and the urldate, rather than just the former. See sections 4.4.2 and 5.4.2.

- After a bug report from Jon Arnold, I have tried to fix the field-inheritance schemes for audio, music, and video entries, which should now work more intuitively in all styles if you use crossref fields in them. Concretely, this means that a title in one such entry (identifying a complete work or collection) can become a booktitle in the child entry, identifying the work of which the title is only one part.

- Tobias Becht suggests that using specialized citation commands for printing author-only citations of ancient works (cf. *Manual* 14.244–5) is unnecessarily awkward. I have therefore provided the notitle option (sections 4.4.3 and 5.4.3) to turn a short author-title citation (whether in the notes & bibliography style or the author-date styles) into an author-only citation. You can set this option by entry type or in the options field of individual entries.

- Tobias also pointed out that the *Manual* permits using short citations of ancient, medieval, and Renaissance works right from the first citation in the notes & bibliography style, even if you don't want to do the same for the entire document. You can therefore now set the package option short for the whole document, for specific entry types, and for specific entries.

182

- Again at Tobias' prompting, I've modified how the ibidem mechanism works for entries with classical entrysubtype. This includes following the *Manual's* guidelines about suppressing the *ibid.* string when using abbreviated forms of authors' names (unless you've set useibid=true in the options field of the entry itself). I've also made the noibid and useibid options more granular, the former in all styles and the latter in notes & bibliography. You can now set them globally, per type, and per entry.

- Having noticed that the journaltitleaddon field is standard in biblatex, I have added it to the article and review types, as well as adding the titleaddon field to the periodical type, and this in all styles. The jtitleaddon option controls the separator between the main title and the annex in all these cases. It is set to space by default. See sections 4.4.2 and 5.4.2.

- Alanna Warner-Smith reported that the author-date styles weren't distinguishing citations with extradate letters when a journaltitle replaced the author. This is, I hope, fixed.

- After input from Claudius Ellsel, I've revised the hyperref functionality in the author-date styles, allowing you to choose which parts of a citation will act as links to the entry in the reference list. Two new user options, hyperall and hypername, complement the pre-existing hypertitle to give more fine-grained control. All of these options are available globally, per type, and per entry. See section 5.4.3.

- After a request from Samuel Foster, I've provided the suppressnoterefs option for the notes & bibliography style, settable per entry type and per entry, which prevents the printing of the noteref back reference after such entries. This may be more convenient than using the specialized citation commands to achieve the same end. Cf. section 4.4.4.

- There were many other bugfixes, including another attempt to make the postnotepunct option work more reliably, improvements to shorthand behavior in the author-date styles, and fixes, suggested by David Thiel, to the hyperref functionality when using cmsendnotes.sty.

2.2: Released June 30, 2021

All styles still require the current biblatex (3.16) and biber (2.16). The next release of biblatex (3.17) will necessitate some changes to Chicago's date handling, so a new release of biblatex-chicago will, I hope, follow swiftly on that of biblatex.

- Marçal Orteu Punsola has very generously provided a Spanish localization for biblatex-chicago (cms-spanish.lbx).

- I have modified several other localization files, fixing typos and ensuring minimal harmony when Chicago provides the short string (in notes) and standard biblatex the same string in long form (in the bibliography).

2.1: Released March 27, 2021

This release contains several new features, some modifications for compatibility with the latest biblatex, and many bug fixes. All styles require the latest biblatex (3.16) and biber (2.16).

- Patrick Danilevici has very generously provided a Romanian localization for biblatex-chicago (cms-romanian.lbx) to go with the romanian.lbx he wrote for biblatex itself. The only hitch is that the latter file won't be in the standard biblatex package until the next release, so you'll have to grab it from GitHub until then.

- Moritz Wemheuer recommended that I include the standard biblatex macro begentry at the start of all drivers in both Chicago styles, with an empty definition by default. This allows users to redefine it in their preambles and thereby inject their code into the driver before anything is printed.

- Moritz also drew my attention to a question on StackExchange which suggested that I should expand the applicability of the annotation field, so I've modified the annotation option in all styles to allow the field to appear both in bibliography (reference list) and in long notes, and also added two new options in all styles, bibannotesep and citeannotesep, which allow you to define how the annotation field will relate to the main entry. The options have different defaults, and although they have the same pre-defined keys those keys may have slightly different meanings depending on whether the annotation is to appear in notes or bibliography. Please see the docs of field and options in sections 4.2, 4.4.2, 4.4.3, 5.2, 5.4.2, and 5.4.3.

- On a related subject, I have included two new options in all styles — formatbib and entrybreak — which provide fine-grained control of page breaking in bibliographies and reference lists. Some of the values of these options are designed to be used with specific values of the bibannotesep option, so please see the documentation in sections 4.4.3 and 5.4.3.

- In the notes & bibliography style Fr. Norbert Keliher requested a way to turn off the printing, in the first citation of a work, of the introduction — (hereafter cited as ...) — of a shorthand that will appear in subsequent citations. The shorthandintro option is now available globally, per type, and per entry to allow you to control the (non)appearance of this clarifying notification. By default, the first citation of a work will continue to present the shorthand as it always has. Please see section 4.4.3 for the details.

- Philipp Immel requested a feature involving the nameaddon field, and pointed to a discussion on StackExchange which suggested that others were also unsatisfied with that field's functionality, so in this release I've provided three new package options to allow users to mould it to their needs. The options are called nameaddon, nameaddonformat, and nameaddonsep, and you can find all the details in the nameaddon docs in sections 4.2 and 5.2, above. Cf. also the discussions of the options in sections 4.4.2, 4.4.3, 5.4.2, and 5.4.3. Allow me to emphasize that the package defaults remain exactly the same as before, so that if you don't set any of the new options no changes to your documents or .bib databases are necessary.

- Philipp Immel also requested a way to use the standard biblatex option datamodel even when loading the Chicago styles with \usepackage{biblatex-chicago}. Implementing this allowed me to regularize how the standard mcite and natbib options were passed through to biblatex, and also to support the new biblatex casechanger option, which can help solve a bug reported by Peter Mukunda Pasedach. See sections 4.4.3 and 5.4.3, above.

- The new relatedtype short is available in all styles to allow the presentation of short references rather than full ones when using the related mechanism. See sections 4.2.1 and 5.2.1.

- In a similar vein, I have provided new \citeincite and \fullciteincite commands to allow you, in some contexts, to cite a related work inside the fields of another entry, something which, although not recommended, is common practice, and which has frequently not worked quite right. The \citeincite and \citeincites commands always present short citations, and are available in several fields, while \fullciteincite and \fullciteincites always present long references, and are designed only to be used in the annotation field for cases where an annotated bibliography or reference list includes references to works not cited in the main text. See sections 4.3.2 and 5.3.2 for the details. If I've missed yet further tricky corners in this functionality please let me know.

- After a request by Patrick Danilevici, I have added the four fields eventdate, eventtitle, eventtitleaddon, and venue to the inproceedings, mvproceedings, and proceedings drivers in all styles, for compatibility with standard biblatex usage. See the docs of those entry types in sections 4.1 and 5.1.

- The postnotepunct option should generally work now, both in more contexts and with arbitrary postnote field contents.

- The dashed option governing the use of the 3-em dash in bibliographies and reference lists is now available globally, per type, and per entry.

Note on the 16th-edition files:

- These have been updated for compatibility with the latest biblatex and biber, and there are also a number of bug fixes included. I haven't added any new functionality to the files, so for instance if you want to use the new Romanian localization you'll need the current 17th-edition styles. With this release I am marking the 16th-edition files as deprecated, and will remove them in a future release. Please consider moving your documents to the current specification, where all of my development time is now spent.

2.0: Released April 20, 2020

Converting from the 16th to the 17th edition in your .bib files and LaTeX documents:

- The 17th edition of the *Manual* no longer encourages use of *ibid.* to replace repeated citations of the same work in the notes & bibliography style, preferring instead to use the author's name alone, along with any page number(s). If you wish to continue using *ibid.* in that style, you need to set the new option useibid=true when loading biblatex-chicago in your preamble.

- If you are loading the package the old-fashioned way, with biblatex and the style= option instead of with biblatex-chicago, please be aware that there are two standard packages required by biblatex-chicago that aren't automatically loaded by biblatex: xstring and refname. You'll need to load them in your preamble yourself.

- If you've been using the year field to present decades like 1950s, this will no longer work accurately in author-date citations. The correct way to do so now is to use one of biblatex's ISO8601-2 date specifications in the date field instead, to wit, 195X. Generally, I've tried to make year fields like [1957?] or [ca. 1850] continue to work properly, but here too the best thing to do is to use the new date features and present them like 1957? or 1850~, respectively. This will ensure that both sorting and punctuation work out properly. See table 3, and the date docs in sections 4.2 and 5.2, above.

- If you have been using the nameaddon field to hold time stamps for online comments, then you should put the time stamp into the date or possibly eventdate field, instead, using the ISO8601-2 format implemented by biblatex. See the date and nameaddon field docs in sections 4.2 and 5.2, along with tables 1, 2, and 3.

- Following on from the previous item, the 17th edition of the *Manual* includes more plentiful and more detailed instructions for presenting online materials than were available in previous editions. For users of biblatex-chicago this means that there is now guidance for many more sorts of material than before, so if you have been improvising citations of this sort of material in previous releases it will be worth checking to see whether there are now clearer instructions available. Tables 1 and 2 summarize the new specifications for the notes & bibliography and author-date styles, respectively. Cf. in particular the new commenton relatedtype in sections 4.2.1 and 5.2.1. Also, the **online** entry type now prints both author and editor (or other editorial role) if they exist, and I've moved the addendum field *before* the url, which fits better with indications in the *Manual.* If you've been using the addendum field to present citations of other entries (as in older versions of biblatex-chicago), please switch to the related mechanism, which works better anyway.

- On the same subject, if you are using the notes & bibliography style and are retaining the crossref field (instead of using the commenton relatedtype) in review entries as a means of presenting comments on blogs, such entries are now subject to the settings of the longcrossref option and will appear in abbreviated form in some full notes and in the bibliography, as has always occurred in incollection entries, for example. You can set longcrossref to true to get back the old behavior.

- The 17th edition generally encourages more strongly than the 16th the use of only one publisher in the publisher field. If you decide to retain more than one, and one of them is a part of an academic publishing consortium, it encourages you to specify this relationship rather than merely listing the two using the keyword "and" in the field. Please see the documentation of the field in sections 4.2 and 5.2, above, for the rather minor (and rare) changes this might mandate for your .bib files.

- The 17th-edition presentation of **music** entries has added a few pieces of information it seems to find desirable — track number in chapter and specification of a lead performer's role in, e.g., editortype — though the basic structure of a 16th-edition .bib entry remains unchanged. Please see the documentation of that entry type in sections 4.1 and 5.1, above.

- The 17th edition has added a couple of wrinkles to the **video** type specifications. You can now put the broadcast network of a TV show in the usera field, and you can also supply the new entrysubtype tvepisode to print the series title (booktitle) *before* the episode title (title). Please see the documentation of the entry type in sections 4.1 and 5.1, above.

- Both Chicago styles now sentence case the title field in **patent** entries, so you may need to put curly braces around words that shouldn't appear in lowercase.

- The pubstate field now has a more generalized functionality, while maintaining the specialized uses present in earlier releases. In particular, please note now that almost anything you put in the field will be printed somewhere in the entry, and in the case of the author-date styles may appear in a somewhat different part of the entry than that to which you may have become accustomed. If you wish to move this data back to the end of the entry in the author-date styles, then the addendum field might be of service. The documentation in sections 4.2 and 5.2 should help.

- Biber is now the *required* backend for all Chicago styles, including the 16th-edition files still included in the package. If you have somehow been using some variant of BIBTeX in the notes & bibliography style up to now, I'm fairly confident it will no longer work. Please switch to biber.

Other new features common to the notes & bibliography and author-date styles:

- Wouter Lancee has very generously provided a Dutch localization for biblatex-chicago, called cms-dutch.lbx. You can use it by including "dutch" when loading babel. Gustavo Barros has also very kindly provided a much-revised version of his cms-brazilian.lbx.

- As mentioned above, this release for the first time implements biblatex's elegant and long-standing support for the ISO8601-2 Extended Format date specification, which means there are now greatly enhanced possibilities for presenting uncertain and unspecified dates and date ranges, along with date eras, seasons, time stamps, and time zones. Table 3 summarizes the implementation for all Chicago styles, but see also the date field in sections 4.2 and 5.2, along with the new package options alwaysrange, centuryrange, decaderange, nodatebrackets, and noyearbrackets.

- I have also implemented year-range compression in all styles, governed by the new compressyears option, set to true by default.

- Constanza Cordoni requested a way to turn off the printing of the 3-em dash for repeated names in the bibliography or reference list, and the *Manual* concedes that some publishers prefer this style. Some of biblatex's standard styles have a dashed option, so for compatibility purposes I've provided the same. By default, I have set it to true to print the name dash, but you can set dashed=false in your preamble to repeat names instead throughout your document.

- Jan David Hauck suggested I extend the field-exclusion functionality beyond the package options already provided (sections 4.4.2 and 5.4.2) by biblatex-chicago. First,

I made sure that all of those options could be set globally, per type, and per entry. Second, I added the command `\suppressbibfield`, designed to appear in the preamble, and which will look something like:

```
\suppressbibfield[entrytype,entrytype,…]{field,field,field,…}
```

Please see sections 4.3.1 and 5.3.1 for the details.

- After a request by user BenVB, I have added support for the biblatex **shortseries** field, which allows you to present abbreviated series in book-like entries in all the styles. You can use the `seriesabbrev` option to control where in your document these abbreviated forms will appear. By default, the field is ignored in all styles. You can also print a list of series abbreviations, rather in the manner of a list of shorthands, using a command like: `\printbiblist{shortseries}`. Please see shortseries in sections 4.2 and 5.2.

- I have added a new preamble option, `cmsbreakurl`, which attempts to follow the *Manual's* instructions for line-breaking inside URLs. It doesn't work 100% accurately, and it doesn't play well with the `ragged2e` package, but in most circumstances it is at least closer to the Chicago ideal than the biblatex defaults. See sections 4.4.3 and 5.4.3.

- The *Manual* now specifies how to present articles with a unique numeric or alphanumeric ID, which you can place in the `eid` field. If you've been using this field in previous releases you'll notice that the ID has moved to a different place in long notes, bibliography, and list of references.

- In jurisdiction entries, the presentation of the title changes depending on whether it appears in a stand-alone citation or as part of the flow of running text, no matter whether the citation is in the main body or in a note. I have provided the `\runcite` command, in both Chicago styles, for jurisdiction citations that appear in running text.

- N. Andrew Walsh suggested that I allow editorial roles that aren't pre-defined `\bibstrings` to appear as-is in entries, just as the standard biblatex styles do. I have followed this advice for the `editortype`, `editoratype`, `editorbtype`, `editorctype`, and `nameatype` fields, making sure to capitalize the string according to its context.

- I have added the venue field to **misc** entries, both with and without an `entrysubtype`. It also appears in the new **performance** type.

- I have added the version and type fields to **book** entries to help with multimedia app content (14.268). This material fits quite well in such entries but needs extra fields to present information about the version of the app and also the system type on which it runs. I added the type field to **article**, **review**, and **online** entries for presenting medium information for online multimedia (14.267).

- I have added a new entry type, **dataset**, to allow the citation of scientific databases. Cf. sections 4.1 and 5.1.

- I have added the number field to **misc** entries with an `entrysubtype` to help cope with the varieties of location information in different archives.

- The new entry type **standard** is now available to cite standards published by national or international standards organizations. If you have been using the book type for such material it might be worth switching to make sure your entries are more in line with the *Manual's* specifications. See the docs in sections 4.1 and 5.1 for the details.

- The new entry type **performance** is now available for citing live performances. You can sometimes also use a **misc** entry without an `entrysubtype`.

- I have added the eventdate field to the **audio** entry type to help it cope with the presentation of podcasts, which are new to the 17th edition. Please see the documentation of the entry type in sections 4.1 and 5.1, above.

- I have added the origdate, eventdate, userd, and howpublished fields to the **artwork** and **image** entry types, in response to additional information given in some of the *Manual's* examples. Please see the documentation of **artwork** in sections 4.1 and 5.1, above.

- I have added the maintitle, mainsubtitle, and maintitleaddon fields to the **article**, **periodical**, and **review** entry types, where it (they) will hold the the name of any larger (usually periodical) publication of which a blog is a part. This departs from standard biblatex usage, but the need for two italicized titles demanded something like it.

- I have added a new field-exclusion option, urlstamp, set to true by default, which means that any time stamp associated with the urldate will always be printed. You can set it to false in the preamble either for the whole document or for specific entry types, or in the options field of individual entries. See the docs in sections 4.4.2 and 5.4.2, above.

- The howpublished field has accumulated a series of new functions in various entry types, bringing it far from its origins in booklet, misc, and unpublished entries. Please see its documentation in sections 4.2 and 5.2.

- In inreference, mvreference, and reference entries biblatex-chicago no longer considers any of the name fields (author, editor, etc.) for sorting purposes in the bibliography or reference list, thus leaving the title as the first field to be considered. This may simplify the creation of .bib database entries.

New notes & bibliography features:

- In keeping with indications in the 17th edition of the *Manual*, I have provided a means for altering the syntax when presenting multi-volume works, i.e., for presenting the title of the whole series (maintitle) *before* the title of individual volumes of that series (title or booktitle). This involves the use of the new relatedtypes maintitle and maintitlenc, which may be used in bookinbook, inbook, incollection, inproceedings, letter, mvbook, mvcollection, mvproceedings, and mvreference entries. Please see the detailed documentation of this feature in section 4.2.1, s.v. relatedtype maintitle.

- I have implemented a new system of back references from short notes to long notes to help readers find fuller information about a source more quickly and conveniently, as envisaged by the Manual. The feature is enabled with the noteref option, and there are several sub-options to control where and what is printed: fullnoterefs, noterefinterval, noterefintro, pagezeros, hidezeros, and endnotesplit. The dependent package cmsendnotes.sty can assist if you use endnotes instead of footnotes in this context. It too has numerous options: hyper, enotelinks, noheader, blocknotes, split, subheadername, headername, runningname, and nosubheader, alongside the new commands \theendnotesbypart and \cmsintrosection. Four new citation commands complete the provisions: \shortrefcite, \shorthandrefcite, \shortcite*, and \shorthandcite*. Please see section 4.4.4 for all the details, and also cms-noteref-demo.pdf for a brief example and explanation of some of the functionality.

- I have ported, with modifications, the author-date package option nodates to the notes & bibliography style. It is set to true by default. In conjunction with the nodatebrackets and noyearbrackets options it provides an alternative presentation of uncertain dates. See section 4.4.2.

- Pursuant to a bug report by David Purton, I have recoded the various \headlesscite commands and included a new one, \Headlesscite, which is the actually functional way to enforce capitalization at the start of such a citation, should you need to do so.

New author-date features:

- The **verbc** field, which is standard but unused in the styles included in biblatex, allows the user fine-grained control over if and when an extradate letter (1976**a**) will appear after the year in citations and the list of references. See its documentation in section 5.2.

- The new authortitle type and entry option allows you to provide author-title citations in the text instead of author-date. The entrysubtype value classical does the same, but there may be cases where using such an entrysubtype is impossible. This is set to true by default for **dataset** entries.

- On the same subject, you can also use the new citation commands \atcite and \atpcite to achieve the same end. The former prints a plain citation, the latter places it in parentheses.

- In the default configuration, when you use a shorthand field the style will now sort properly by that field, which is the first thing to appear in reference list entries. If you set cmslos=false in your preamble then this no longer applies, as the shorthand no longer appears in the reference list.

Note on the 16th-edition files:

- These have been updated for compatibility with the latest biblatex and biber, and there are also a number of bug fixes included, many of them already mentioned in changelog items above. The compressyears option is available and turned on by default, and so is the dashed option. Most of the new ISO8601-2 Extended Format date specifications are available, also, though time stamps won't be printed, as that edition of the *Manual* is mostly silent about them.

1.0rc5: Released January 16, 2018

- As Nikola Lečić spotted, recent releases of biblatex have introduced some compatibility problems for biblatex-chicago, particularly with regard to the handling of the origlanguage field (now a list), but also through the renaming of several other fields and declarations (e.g. \DeclareSortingTemplate). I have improved the handling of the origlanguage list by including many new bibstrings in the package's localization files, but other changes to formatting macros have made backward compatibility with older releases of biblatex difficult or impossible. Please upgrade to version 3.10 — which has received the most testing — to use these styles.

- As I mentioned in the Notice (section 1), the 17th edition of the *Manual* has now appeared, and my development energies from this point will be devoted to upgrading all styles to conform to it. You can still file bug reports against the 16th edition, but the next major feature release will be based on the 17th. In preparation for these changes, I have removed all the 15th-edition files from the package.

Other New Features:

- After fielding multiple requests over the years, I have added three new entry types — **jurisdiction**, **legal**, and **legislation** — to allow the presentation of court cases, laws, treaties, congressional (parliamentary) debates and hearings, constitutions, and executive documents. The first (round and square) and last (canada, constitution, hansard, hearing, uk, and un) introduce a number of new entrysubtypes to help with formatting quirks, including the presentation of Canadian and UK materials for inclusion in an otherwise US context. There are also several new options (legalnotes, noneshort, short, and supranotes) for controlling the output. I have documented all of this in section 6 above, a separate section both because the specification really comes from the *Bluebook* rather than the *Manual*, and also because they are the only entry types treated identically by the notes & bibliography style and the author-date styles (itself a formatting quirk). You can also look at the sample files legal-test.bib and cms-legal-sample.pdf to see how you might construct your database entries. Support for *Bluebook* citations is in its infancy, so if you have

ideas for sorting out its complexities more elegantly or spot any inaccuracies then I would be happy to hear about it. The implementation is intended mainly for American documents, but there is some rudimentary localization for the other languages supported by biblatex-chicago. The actual citations in such contexts would, let it be noted, fall outside of the *Bluebook* spec.

- I am grateful to Gustavo Barros for providing a Brazilian Portuguese localization for biblatex-chicago, contained in the cms-brazilian.lbx file.

- Gustavo also pointed out some instances where the package's bibstrings couldn't accommodate the needs of his localization, so with his help I've split the recorded string into discrecorded and songrecorded, then added it to all the .lbx files. I've also added two new bibstrings for the lista field format: subverbo and subverbis. I've added them to all the .lbx files, but only cms-brazilian.lbx differs from the default. If other languages need this change please let me know.

- The same user also suggested a fix to **patent** entries: removing the comma from between the dates when the language doesn't use a comma in lists.

- Timo Thoms pointed out some annoying inconsistencies when using the hyperref package with the author-date styles, and I have attempted to rectify them. In citations, only the date portion should act as a link, if there is a date, otherwise a title or perhaps a shorthand will link to the entry in the list of references. If you have entries that you believe should present hyperlinks but don't, you can try setting the new hypertitle option in their options fields. Alternately, you can set the option to true globally in the preamble and then titles and shorthands will serve as links whether there's a date or not. Cf. section 5.4.3, above.

- Bertold Schweitzer requested that the styles allow using the string forthcoming in the pubstate field to present sources that are yet to be published. This is now supported in all styles, and has the additional benefit of rendering recourse to the \autocap command unnecessary, as the styles print \bibstring{forthcoming} where the year would normally appear. Using the year field itself is, of course, still supported too.

- The same user requested that I allow newspaper as an exact synonym of magazine in the entrysubtype field of article, review, periodical, and suppperiodical entries. I have provided this in all styles, and wherever you see magazine in this documentation then newspaper will work in exactly the same way.

- Bertold also suggested that, following the example of Philip Kime's biblatex-apa package, I support the use of related functionality when presenting reviews, so that you can, for example, easily present multiple reviews of the same item. I have provided this functionality in all styles. To enable it you'll need to set the relatedtype field to reviewof in article, review, or suppperiodical types. You should also read the documentation in section 4.2.1 or 5.2.1, above, as this relatedtype works somewhat differently from the others. The standard, manual way of citing such works remains, of course, available.

- Jan David Hauck suggested that there was a need for an unpublished entrysubtype to the report type, which would present the title in quotation marks (or plain roman in authordate-trad) instead of italics. I can't quite tell if the *Manual* agrees, but I have fulfilled this request in all styles.

- The same user pointed out that standard biblatex and the discussion in the *Manual* both suggest providing venue, eventdate, eventtitle, and eventtitleaddon fields for the unpublished type, thereby allowing for the further specification of unpublished conference papers and the like. I have added these fields in all styles.

- At the request of N. Andrew Walsh, the notes & bibliography style now offers a way to disambiguate references to different sources which would ordinarily produce identical short notes, that is, where the author and labeltitle are the same. Biblatex's uniquework option is now active by default, and biblatex-chicago provides

three new user options, one for choosing a disambiguating field, one for setting the punctuation between that field and the rest of the short note, and one for formatting the field using parentheses or square brackets — shortextrafield, short-extrapunct, and shortextraformat, respectively. Please see section 4.4.3, above, for the details, and note that shortextrafield has to be set for the mechanism to print anything at all.

- User Pétùr spotted two long-standing bugs: first, that the url=false option didn't stop the printing of the urldate, and second that empty parentheses would appear in some circumstances around non-existent dates in the author-date styles. I have fixed both.

- Philipp Immel wondered whether I could address a long-standing bug when presenting a subtitle after a title that ends in an exclamation point or question mark. This bug has existed since the first release of the 16th-edition styles, and I think I've finally solved it now after the release of the *Manual's* 17th edition. (Cf. batson.)

1.0rc4: Released May 2, 2017

Another bug-fix release.

- Marko Wenzel reported, and helped to fix, a fairly major problem with the date handling in the author-date styles, an issue I hadn't spotted when doing the date-related updates for 1.0rc2.

- I've also fixed a long-standing inaccuracy in the date-handling code of **patent** entries in the author-date styles. Such entries now behave as the documentation claims they do.

1.0rc3: Released April 20, 2017

This is a minor bug-fix release.

- Charles Schaum reported a whitespace bug that appeared when using multiple languages with Babel. This was introduced in the last release by some careless editing by me, and should be fixed now.

- Charles also pointed me to a discussion about a problem using BIBTEX with biblatex-chicago. Ulrike Fischer very kindly suggested an elegant solution, and I have integrated it into this release.

1.0rc2: Released March 26, 2017

This is an interim release designed mainly to fix a number of subtle issues, pointed out by several users, that appear when you use the newest version of biblatex (3.7). These were mostly concentrated in the date-handling code, which I believe now behaves correctly, and should do both with the newest biblatex and with somewhat older releases. A much larger set of new features is still pending, but I have fixed some other bugs and added a few new options:

- J. P. E. Harper-Scott pointed out that, in ordinary British usage, day numbers are presented as plain cardinals rather than ordinals. The *Manual* itself also prefers this format, not only for American-style dates but also for British ones, so I think the previous behavior of the package was a bug. I have in both styles set the default presentation of British day numerals to be plain cardinals, providing a new preamble option ordinalgb restoring the previous default and printing ordinal dates when using the british language with Babel.

- I have, in both styles, attempted to provide an improved \partedit macro, the old one being inconvenient for users writing in French. The new macro should work now without manual intervention to provide the correct form of the preposition (de or d'). If you are using the french option to Babel, please take care to remove any hand-formatting you might have provided in these contexts.

- Jan David Hauck has both reported a bug in the `\gentextcite` code in the author-date styles and also pointed me to its solution, as provided by moewe on [Stackex-change](). It turned out there were other bugs in that code, now also fixed.

- User laudecir requested a way to present a shorthand even in the first citation of a given work. The new `shorthandfirst` option in the notes & bibliography style can be set to `true` either in the preamble or in individual entries, and should make this functionality simpler to activate than the `\shorthandcite` command.

- Also in the notes & bibliography style, Stefan Björk requested a way to turn off the printing of url, doi, and eprint information in notes but not in the bibliography. The new `urlnotes` option, which you can set to `false` in the preamble or in individual entries, provides this. Please note that it does not apply to online entries.

- Several users pointed out the presence of warnings in .log files caused by deprecated grammar in the default Sorting Schemes of both styles. These should now be fixed.

1.0rc1: Released June 7, 2016

Obsolete and Deprecated Features:

- The 15th-edition styles are now obsolete, and have been moved to a new `obsolete` subdirectory. You can still use them as they stand, but they won't compile against the newest biblatex, so you'll have to make sure that you have an older version (2.9a, perhaps). If you are still using them, I strongly urge you to consider switching to the the 16th-edition styles, which contain many new features and bug-fixes.

- The old Chicago-specific option `usecompiler` is deprecated, and has been replaced by the standard biblatex `usenamec`. If you have been using the former in your preamble or in your .bib entries, please replace it with the latter, which works better across the board. Usecompiler still "works," just not very well.

Other New Features:

- Stefan Björk has very generously provided a Swedish localization file for the package — `cms-swedish.lbx` — which can be loaded and used with babel just like the other localizations.

- I have added support for **related** functionality to all the Chicago styles, including all the standard biblatex **relatedtypes**. It is turned on by default in all styles, but you can turn it off, or alter where the information is printed, using the `related` option in the preamble or in individual entries. In the notes & bibliography style, related information is printed by default only in the bibliography, but you can change that by setting the option. In the author-date styles, it will only ever print in the list of references, depending on the option's setting. Please see sections [4.2.1]() and [5.2.1]() for the details.

- I have improved the name-handling code in all styles, regularizing the functioning of the `namea`, `nameb`, and `namec` fields with respect to the other, standard biblatex names. The former two in particular are newly available in the collection and periodical entry types, and biblatex-chicago now recognizes the standard `usenamea`, `usenameb`, and `usenamec` toggles, the last replacing the deprecated `usecompiler` (as above). You can also now use the **nameatype** field just as you would an editortype, extending the possibilities for identifying certain roles attached specifically to titles as opposed to booktitles or maintitles.

- After a request by user BenVB, I have added support for the biblatex **shortjournal** field, which allows you to present abbreviated journaltitles in all the styles. You can use the `journalabbrev` option to control where in your document these abbreviated forms will appear. By default, the field is ignored in the notes & bibliography style, and appears only in citations in the author-date styles. You can also print a list of journal abbreviations, rather in the manner of a list of shorthands, using

a command like: `\printbiblist{shortjournal}`. Even though the periodical entry type uses the title and shorttitle fields in place of journaltitle and shortjournal, these entries are included in this functionality, and controlled by the same journalabbrev option. Please see s.v. "shortjournal" in sections 4.2 and 5.2.

- Following a request by Arne Skjærholt, and his generous provision of some code to get me started, I have implemented a new `\gentextcite` citation command in all styles. The "gen" part of the name refers to the genitive case, and it adds a possessive ending — **'s** by default — to the author's name in what is otherwise an ordinary `\textcite`. You can change the added ending however you want, using a third optional field to the citation command, and you can control to which names the ending is added in a `\gentextcites` multicite command by using the genallnames preamble and entry option. Please see sections 4.3.2 and 5.3.2 for the details.

- Stefan Björk pointed out that url, doi, and eprint information could be totally ignored in some entries when you used the abbreviated cross-referencing functionality accessed through the crossref and xref fields. At his request, I have provided a new xrefurl entry and preamble option for all the styles to control the printing of this information in abbreviated notes or bibliography (reference list) entries. Please see sections 4.4.3 and 5.4.3 for the details.

- In a related change, I have stopped child entries inheriting url, doi, and eprint fields from their cross-ref'd parents, so if your documents rely on this behavior please note that you'll have to provide such fields manually in the child entries.

- Roger Hart long ago requested a way to control the punctuation before book-, main-, or plain titleaddon fields, and I have finally added it in this release in the form of two entry and preamble options, ptitleaddon and ctitleaddon, available in all styles. By default, the former prints `\addperiod\addspace`, hence its name, and the latter `\addcomma\addspace`, but you can change either or both depending on which field you are using and which sort of entry it appears in — the default output can be your guide to which option(s) to change. Please see the available valid option keys in sections 4.4.2 and 5.4.2.

- The same user also long ago requested that the notes & bibliography style make it possible to use *Idem* when two consecutive notes cite different works by the same author. You can now use the standard biblatex option idemtracker=constrict in your preamble to activate this in your documents, but please be aware, first, that the *Manual* doesn't exactly approve of this and, second, that you'll only see *Idem* in short notes, never in full ones, which seems to be the standard (biblatex) way of implementing this.

- Also only in the notes & bibliography style, I have added a shorthandpunct option to control the punctuation that appears before the first appearance of a shorthand and/or a shorthandintro in a long note. The default is `\addspace`, but you can change it in your preamble or in individual entries. Please see the available valid option keys in section 4.4.2.

- After reading a discussion started by Ryo Furue at github, I have added, in the author-date styles only, a test to some spacing commands to prevent line breaks immediately after abbreviation dots. These tests apply only in running text, never in the list of references, where good line breaks are already hard enough to find.

- In addition to moving all the 15th-edition styles into the obsolete subdirectory, I have also reorganized the author-date style files, adding chicago-dates-common.cbx which contains the code that is common to the trad and the standard authordate styles. Nothing has changed in terms of loading the styles, the changes being designed primarily to ease maintenance.

- I have created two new documentation files (and an appendix) to provide short introductions to the Chicago styles, introductions which attempt to fill the gap between the Quickstart section (2) and the fuller documentation contained in sections 4 and 5 of this file. Both cms-notes-intro.pdf and cms-dates-intro.pdf are fully

hyperlinked so you can move easily from formatted citations and (annotated) references to .bib entries and back, with marginal references to the fuller discussions here. There is also a short cms-trad-appendix.pdf file to discuss a few entries that would need special treatment for the trad style. The sample files for each style still exist, but I intend them mainly for testing purposes, while many more (annotated) entries are still available for consultation in notes-test.bib and dates-test.bib.

- I have made a number of other small enhancements to and fixed numerous bugs in all the styles, including some subtle inaccuracies in author-date citations spotted by Arne Skjærholt and some macros in inproceedings entries that had been missing for years. I have provided some default values for counters in biblatex-chicago.sty that aid in breaking long urls across lines, but I make no pretense that these fully adhere to the *Manual's* specifications. I have added a few bibstrings, currently missing in standard biblatex, to cms-german.lbx for use with the related functionality. Recommendations for better ones would be gratefully received.

0.9.9i: Released May 16, 2016

- This is another interim release, allowing the use of biblatex 3.4 for those who want to try it. I have also fixed one old formatting error when "n.d." appears in author-date citations. A full feature-release based on 3.4 is imminent.

0.9.9h: Released March 22, 2016

- This is an interim bug-fix release, updating the styles so that they will work with biblatex 3.3. The notes & bibliography style, as pointed out by several users, wouldn't compile at all with the newest biblatex version, and all styles had inaccuracies in the presentation of names due to changes in the name-handling code in biblatex. I've done some testing against biblatex 3.3, and fixed all the errors I've spotted, but there may still be parts of my code that need updating to work well with the current version, so you can still downgrade to an earlier biblatex — I recommend 2.9a — if 3.3 doesn't work for you. The next release will be a feature release, so if you've made a request, it should be fulfilled then.

- I've also fixed a couple of long-standing bugs, one in the entry options controlling abbreviated cross-references and another in the formatting of the prenote field, the latter identified (ages ago) by Bernd Rellermeyer.

0.9.9g: Released August 21, 2014

- Alexandre Roberts found a showstopper in the functionality related to the new inheritshorthand option in the notes & bibliography style, and I found an unpleasant bug in the formatting of abbreviated cross-references in the same style. This release, I hope, fixes both, but is in all other respects identical to 0.9.9f.

0.9.9f: Released August 15, 2014

- I've made the alterations needed to bring the styles into line with the latest version of biblatex (2.9a). This is the version that has been tested most thoroughly with biblatex-chicago, so I strongly recommend using it.

- I fixed several inaccuracies in the presentation of abbreviated cross-references in all the Chicago styles, and while I was working on that portion of the code it seemed an opportune moment to fulfill some feature requests bearing on the same area of functionality.

- First, following a request from Alexandre Roberts, I have added the inheritshorthand option to the notes & bibliography style, which allows child entries to inherit the shorthand field from their parents. This in turn allows the shorthand itself to appear in place of the usual abbreviated citation of parent entries cross-referenced by several different child entries, thereby saving some space. (This behavior was already available in the author-date styles, so the option is unnecessary there.) You'll need to use skipbiblist in the options field of child entries to make the list of shorthands work correctly. Please see the documentation of the shorthand field for the full explanation.

- Second, following a request from Kenneth Pearce, I have added to all Chicago styles the capacity to combine abbreviated cross-references with the presentation of the original text of translations (via the userf field) or of the original publication details of an essay or chapter you are citing from a subsequent reprint (via the reprinttitle field). See the documentation of those fields, and also of crossref, and note that you can now, taking certain precautions as outlined in the shorthand docs, combine the userf, crossref, and shorthand fields. This mechanism contains a great many moving parts, so please report any problems you might have with it.

- Third, and finally, following a bug report by Mark van Atten I have fixed all Chicago styles so that the biblatex backref mechanism works properly in biblatex-chicago, including in those entries that use abbreviated cross-references, and in those that use the userf or reprinttitle fields. I can't see any instructions concerning this in the *Manual*, so I've left the formatting of backref lists in the hands of biblatex itself. If the default behavior doesn't match your needs, let me know, as it's possible I could add some further options for modifying it.

- I have added a new compresspages option to all the Chicago styles. If set to true it automatically compresses page ranges in the pages and postnote fields, allowing you to type ranges naturally, e.g., 101--109, and letting the package follow the *Manual's* rules for you. (In this case, it would yield 101–9 in the document.) Thanks are due to David Gohlke who brought to my attention a discussion that took place a couple of years ago on Stackexchange regarding the automatic compression of page ranges. Biblatex has long had the facilities for providing this, and though the *Manual's* rules (9.60) are fairly complicated, Audrey Boruvka fortunately provided in that discussion code that implements the specifications. As some users may well be accustomed to compressing page ranges themselves in their .bib files, and in their postnote fields, I have made the activation of this code a package option.

- Several users, most recently David Gohlke, have requested a way to alter the punctuation that appears just before the postnote argument of citation commands. This allows, in the notes & bibliography style, citations to fit better into the flow of text, while in the authordate styles it allows you very easily to insert comments, which follow a semi-colon, inside parenthetical text citations. This punctuation is a complex issue in the *Manual*, but as a first stab at enabling this greater flexibility, I have introduced the postnotepunct package option. Set to true, it allows you to start the postnote field with a punctuation mark (.,;:) and have it appear as the \postnotedelim in place of whatever the package might otherwise automatically have chosen. Please note that this functionality relies on a very nifty macro by Philipp Lehman which I haven't extensively tested, so I'm labeling this option experimental. Note also that the option only affects the postnote field of citation commands, not the pages field in your .bib file. Note, finally, that if you are using the new compresspages option then any postnote field starting with a punctuation mark will require you to do the compression of page ranges yourself.

- I've added a new inheritance declaration so that **incollection** entries can inherit from **book** entries the same way they inherit from **mvbook**.

- I've fixed a fair number of other bugs, including two in the *Ibidem* mechanism identified by Bernd Rellermeyer, one in the printing of dates, and one in the \textcite command in the notes & bibliography style, these last two pointed out by Kenneth Beesley. The presentation of all the periodical entry types (without an entrysubtype) has also been made more accurate.

0.9.9e: Released January 29, 2014

- This minor release fixes a regression in the *Ibidem* mechanism in the notes & bibliography style, spotted by Harold Bellemare, and present in the package since version 0.9.9c. In all other respects this release is identical to 0.9.9d.

0.9.9d: Released October 30, 2013

- Following requests by Kenneth L. Pearce and Bertold Schweitzer, I have modified and extended the mechanism for creating abbreviated citations when several parts of the same collection are included in a reference apparatus. To the **incollection**, **inproceedings**, and **letter** entries of previous releases, I have added **inbook**, **book**, **bookinbook**, **collection**, and **proceedings** entries. Only **inbook** entries join the former three in having this functionality turned on by default — if you don't want this, it will require intervention either in the preamble or in the options field of individual entries. This intervention will be via the new longcrossref option, which controls the behavior of the four essay-like entry types and defaults to false, while the new booklongxref option controls the four book-like types and defaults to true. The useful settings for the options differ slightly between the author-date and the notes & bibliography specifications, so please see all the details in the docs of the crossref field in sections 4.2 and 5.2, above.

- On the same subject, in the notes & bibliography style, I should mention that in the first, full citation of one part of a collection in a note, the code no longer uses a separate citation of the parent entry to supply parts of what you see printed. (This led to numerous inaccuracies.) If your setup uses a side-effect of the old code to print data that hasn't even been inherited by the child, you may find that you need to change some xref fields to crossref fields to make it work correctly now. This situation will, I imagine, be very rare, but you can look at white:ross:memo in notes-test.bib to see an example.

- In the author-date styles, several users have been frustrated by the lack of an approved way of setting the cmsdate option in the preambles of their documents, and Kenneth L. Pearce requested that I attempt to ease the burden on users by looking at this again. With this release, you can now set cmsdate either to both or on in the preamble, and it will affect all entries (except **music**, **review**, and **video**) with multiple dates. You can still change this setting in the options field of individual entries, but what you won't be able to change there is the new call to \Declare-Labeldate which puts the origdate first in the list of dates when Biber searches for a labelyear to use in citations and in the list of references. If you have been using the switchdates mechanism to get the origdate as the labeldate, your .bib files may need some editing in order to use the new preamble options. Please see the documentation of the date field in section 5.2 above for all the (voluminous) details.

- Following a request by Rasmus Pank Rouland, I adapted new biblatex code in the \textcite(s) commands in all styles to make them fit more elegantly in the flow of text. Upon reconsideration of the commands in the notes & bibliography style, I slightly modified them, but *only* when used inside a foot- or endnote. In this context, by default, for both \textcite and \textcites, you'll now get the author's name(s) followed by a headless *short* citation (or citations) placed within parentheses. You can use \renewcommand in the preamble of your document to redefine the new \foottextcite and \foottextcites commands to change this formatting. See section 4.3.1, above.

- This release includes support, in all styles, for biblatex's multi-volume entry types: **mvbook**, **mvcollection**, **mvproceedings**, and **mvreference**. See sections 4.1 and 5.1.

- If you use Biber, I have added several new inheritance schemes to all styles to make cross-referenced entries work more smoothly: **incollection** entries can now inherit from **mvbook** just as they do from **mvcollection** entries; **letter** entries now inherit from **book**, **collection**, **mvbook**, and **mvcollection** entries the same way an **inbook** or an **incollection** entry would; the namea, nameb, sortname, sorttitle, and sortyear fields, all highly single-entry specific, are no longer inheritable; and the date and origdate fields of any **mv*** entry will *not* be inherited by any other entry type.

- Following a bug report by Henry D. Hollithron, I've added to **unpublished** entries in all styles the possibility of including an editor, translator, etc.

- Thanks to bug reports from Denis Maier and Bertold Schweitzer, I corrected inaccuracies and outright bugs in many entry types in all Chicago styles that appeared when there was a booktitle and not a maintitle or vice versa. This also involved another rewrite of the code handling the volume field and other related fields in all non-periodical entry types that use them.

- On the subject of the volume field, I added a new preamble and entry option, delayvolume, to the notes & bibliography style. In long notes where this data isn't printed before a maintitle, this option allows you to print it *after* the publication information rather than *before* it, as may sometimes help clarify things, according to the *Manual*. This applies to the non-periodical entry types only. See section 4.4.3.

- On the same subject, in all styles, I have added a new preamble and entry option, hidevolumes. This controls whether, in entries where a volume has been printed before a maintitle, any volumes field present will also be printed, in this case *after* the maintitle. By default, this is set to true, so that the volumes field won't appear in such circumstances. See sections 4.4.2 and 5.4.2.

- On the same subject, I have modified, in all styles, the field format for the **part** field, so that if the field contains something other than a number, biblatex-chicago will print it as is, capitalizing it if necessary, rather than supplying the usual bibstring, thus providing a mechanism for altering the string to your liking. I have also decoupled the part field from the volume field, allowing it to be printed even in the absence of the latter, thereby providing a means to refer to segments of a larger work that don't easily fit the established schemes. The iso:electrodoc entry in dates-test.bib shows an example of how this might work.

- There is a new omitxrefdate preamble and entry option in the notes & bibliography style. It turns off the printing of the child's date next to its title in abbreviated book-like entries *only*, in both notes and bibliography. See section 4.4.3.

- Clea F. Rees requested a way to customize the punctuation when a volume and a page number appear together like so: "2:204." You can use \renewcommand in your preamble to redefine the new \postvolpunct command to achieve this, in all styles. If your document language is French, cms-french.lbx redefines this already and prints something like "2 : 204." See sections 4.3.1 and 5.3.1.

- I extended, in all styles, the functions of the userd field, allowing it to modify a date field if it hasn't already been captured by another date specification in the entry. See the documentation of the field in sections 4.2 and 5.2.

- A bug report from Mathias Legrand helped clear up inaccuracies in the presentation of ordinal numbers in all styles.

- For the author-date styles, another bug report by Kenneth Pearce resulted in the addition of the labelyear to the default cms sorting scheme so that more entries in the reference list are sorted properly without further user intervention.

- George Pigman found an odd punctuation-tracking bug in the author-date styles. This has been fixed.

- Marc Sommer found a bug in the presentation of the prenote field in the author-date styles. This has been fixed.

- In the notes & bibliography style, I improved the behavior of abbreviated foot- and endnotes when using the hyperref package.

- I modified the date-presentation code in all the language files (cms-*.lbx) provided by the package. Now, if an entry contains a (*)year and an (*)endyear that are exactly the same, and there aren't any further month or day specifications, then the year alone will be printed. This allows for the clearing of spurious endyears inherited from parent entries.

- I discovered some unpleasant side effects of my arrangement of the .lbx files devoted to Norwegian, and reverted to the arrangement as originally provided by Håkon Malmedal.

0.9.9c: Released March 15, 2013

- Antti-Juhani Kaijahano has very kindly provided a new Finnish localization for biblatex-chicago, called cms-finnish.lbx. As you will see if you look through it, it is still something of a work in progress. If you would like to fill some of its lacunae, please do let me know.

- Following a report by Bertold Schweitzer, I have added the **namea** and **nameb** fields to **article** and **review** entries in all three Chicago styles. As in all the book-like entry types, they allow you to associate an editor or a translator specifically with a title, rather than, in these cases, with an issuetitle. See the docs on these entry types in sections 4.1 and 5.1, above.

- Thanks to another report by Bertold I have, in all three Chicago styles, corrected inaccuracies in the presentation of the **report** entry type. The number now appears immediately after the type, and the type itself is now capitalized properly depending on its context in an entry.

- A third report by Bertold, detailing inaccuracies in the treatment of the **volume** and **volumes** fields in certain contexts, has resulted in a complete rewrite of the presentation of these (and several related) fields in all non-periodical entry types in all three Chicago styles. This won't require any changes to your .bib files, but the output you see may, in some reasonably unusual situations, change. Please let me know if something doesn't look right to you.

- A fourth report by Bertold revealed some inadequacies with multiple date presentation in the two Chicago author-date styles, issues that particularly involved cross-referenced entries. In addition to some general fixes in the code, I have also slightly changed the functioning of the cmsdate=both and cmsdate=on switches. If, and only if, a work has only one date, and there is no switchdates in the options field, then cmsdate=on and cmsdate=both will both result in the suppression of the extrayear field in that entry. See the **date** field docs in section 5.2, above.

- Following a report by Antti-Juhani Kaijahano, I have modified the presentation of author-less **article** and **review** entries in the reference list of both Chicago author-date styles. If such a source had a magazine entrysubtype, the styles would already use the journaltitle at the head of the entry in the list of references, but if there was no entrysubtype the entry would appear in the list date first. Now, in keeping with the *Manual* (14.175), the title will appear first, in both reference lists and in-text citations. See especially under **article** in section 5.1, above.

- Several users have pointed out annoying formatting errors in the styles. Evan Cortens spotted two bugs in the notes & bibliography style, one of which, under various circumstances, introduced extra spaces into long notes and the other of which affected the formatting of the type field in thesis entries. I have fixed both, also applying the latter fix to several other entry types that use the type field. Bertold Schweitzer pointed out a formatting bug with the issuesubtitle field in the author-date style, now fixed. Mark Sprevak reported some spurious spaces appearing in headers and footers when using the titleps package; the culprits were errors in the cms-*.lbx files, now cleaned up.

- I have rectified a number of other errors, in particular making the automatic provision of abbreviated cross-references more robust in incollection, inproceedings, and letter entries, improving the behavior of the postnote field in certain corner cases, fixing bugs in the handling of pagination and bookpagination fields, and slightly altering the placement of the addendum field in book-like entries to bring it closer to the *Manual's* specification. A number of other, smaller improvements should also bring the styles into closer conformity with the specification.

0.9.9b: Released December 6, 2012

- This release contains a new variant of the author-date style, available as the authordate-trad option when loading biblatex-chicago. This provides the traditional, plain, pre-16th-edition Chicago title handling — sentence-style capitalization, absence of quotation marks in article titles and the like — but in all other respects follows the 16th-edition specification, as suggested by the *Manual* (15.45). Remember that the headline package option can be used to turn off the automatic sentence-style capitalization, meaning that titles will appear as presented in the .bib file, at least as far as capitalization is concerned. Please see especially the documentation of **title** in section 5.2, above, for the details.

- I have updated calls to \DeclareLabelname and \DeclareLabelyear in several .cbx files so that the package works correctly with the most recent version (2.4) of biblatex.

- Following a request by Norman Gray, I have included a \textcite (and a \textcites) command in the notes & bibliography style for the first time. Please see section 4.3.2, above, for the details.

- Following a request by Daniel Possenriede, I have added in all three 16th-edition styles a new switch, only, to the doi option, which prints the doi when present and the url only when there is no doi. The package default remains, however, true.

- I am grateful to Baldur Kristinsson for providing an Icelandic localization file for biblatex-chicago, called cms-icelandic.lbx. You'll see if you look through it that it is still something of a work in progress, but it should cover most needs in that language very well. If you would like to fill in some of the gaps please let me know.

- I am also grateful to Håkon Malmedal for providing Norwegian localizations for biblatex-chicago, contained in the files cms-norsk.lbx, cms-norwegian.lbx, and cms-nynorsk.lbx.

- I have added a new British localization (cms-british.lbx) that should make it much simpler for users to produce documents adhering to that tradition. For further details on the usage of all these localizations please see section 7, above.

- Several users have reported a bug that resulted in doubled bibstrings in certain contexts. This happened only when using localizations for which biblatex-chicago didn't have explicit support, and it should now be fixed.

- I have changed the way the 16th-edition author-date styles handle the *Ibidem* mechanism. In the absence of a postnote field you no longer get empty parentheses, but rather a standard in-text citation. If you do have a postnote field, then only that will appear.

0.9.9a: Released July 30, 2012

- I have made a few changes to biblatex-chicago.sty to allow the package to work with the latest version (2.0) of biblatex. In all other respects this release is identical to 0.9.9. If you do use the package with biblatex 2.0, please let me know if there are issues I need to address. Thanks to Charles Schaum for alerting me to some of them.

0.9.9: Released July 5, 2012

Converting 15th-Edition .bib Files to Use the 16th Edition:

Notes and Bibliography Style

- The specification for **music** entries has been significantly altered for the new edition. You no longer need to worry about the ℗ and © signs in the howpublished field, which will be silently ignored, and the pubstate field now reverts to its usual function of identifying reprints or, in this case, reissues. The spec really only requires a record label (series) and catalog number (number), though publisher is still available if you need it. There is a new emphasis, finally, on the dating of musical

recordings, so that the eventdate gives the recording date of a particular song or other portion of a recording, the origdate the recording date of an entire album, and the date the publishing date of that album. Please see the full documentation in section 4.1, above.

- The specification for **video** entries has also been clarified. For television series, the episode and series numbers go in booktitleaddon instead of titleaddon and, as with music entries, the eventdate will hold the original broadcast date of such an episode, or perhaps the recording/performance date of, e.g., an opera on DVD. The origdate will still hold the original release date of a film, and the date the publishing or copyright date of the medium you are referencing. Please see the full documentation in section 4.1, above.

- You should add **customc** entries to provide bibliographical cross-references from multiple pseudonyms back to the author's name.

- In **suppbook** entries, the *Manual* now requires you to provide the page range (in the pages field) for the specific part you are citing, e.g., an introduction, foreword, or afterword.

- In **patent** entries, the *Manual* now prefers sentence-style capitalization for titles, which you'll need to provide yourself by hand.

- When a descriptive phrase is used as an author, you can now omit an initial definite or indefinite article, which will help with alphabetization in the bibliography.

- A DOI is now preferred to a URL, if both are available.

- On the same subject, a revision date (or similar) is preferred to an access date for online material. You can use the new **userd** field to change the string introducing the urldate, which defaults to being an access date.

- Special imprints are now separated from their parent press by a forward slash rather than a comma, so can just be added to the publisher field with the usual keyword and.

- I have implemented a reasonable, less-flexible facsimile of the Biber-only command \DeclareLabelname which should work for those using any backend. It allows biblatex to find a name for short notes outside the standard name fields, including, notably, in the name[a-c] fields. This should reduce the instances where you need a shortauthor field to provide such a name.

- The Chicago-specific setting of another Biber-only command, \DeclareSortingScheme=cms, allows non-standard fields to be considered by biblatex's sorting algorithms, which should reduce the instances where you need a sortkey or the like in your entries. If you aren't using Biber, the package reverts to the standard nty sorting scheme.

Author-Date Style

- All title fields now follow the rules for the notes & bibliography style as far as punctuation, formatting, and capitalization are concerned. Biblatex-chicago-authordate will deal with most of this automatically, but if you have any hand formatting of lowercase letters within curly braces in your .bib file, you will need to restore the headline-style capitalization there. Also, you'll need to be more careful when you provide quotation marks inside titles, remembering to use \mkbibquote so that punctuation can be brought inside nested quotation marks. These revisions will apply particularly to **title**, **booktitle**, and **maintitle** fields.

- The one exception to these rules is in **patent** entries, where sentence-style capitalization of the title is now specified. You'll have to provide this by hand yourself, as in the notes & bibliography style.

- Because of these changes to title formatting, you'll need to observe the difference between **article** and **review** entries, where the latter contain generic, "Review of ..." titles and the former standard, specific titles.

- The presentation of **shorthand** fields has changed. You no longer need to use the **customc** entry type to include cross-references from shorthands to expansions in the list of references. Now, simply using a shorthand field in an entry places that shorthand in citations and at the head of the entry in the list of references, where it will be followed by its expansion within parentheses. The new system will require help with sorting in the reference list — placing the shorthand also in a sortkey should do the trick.

- On the subject of **customc** entries, the *Manual* now recommends using cross-references in several contexts, particularly when a single author uses more than one pseudonym. Adding customc entries makes this happen.

- There have been significant changes when presenting book-like entries with more than one date. If you are using the cmsdate=on option, or indeed simply placing the earlier date in the **date** field and the later one in **origdate**, the presentation will be the same as before, but you should understand that the *Manual* no longer recommends this origdate-only style. It prefers, instead, to present either the date alone or both dates in citations and at the head of reference list entries. When presenting both dates, there is now no longer a choice between the old and new options for cmsdate, but only the both option. If you have old or new in your .bib files, they will be treated as synonyms of both.

- The specification for **music** entries has been significantly altered for the new edition. You no longer need to worry about the ℗ and © signs in the howpublished field, which will be silently ignored, and the pubstate field reverts to its more usual function of identifying reprints or, in this case, reissues. The spec really only requires a record label (series) and catalog number (number), though publisher is still available if you need it. There is a new emphasis, finally, on the dating of musical recordings, which means that such entries will fit better with the author-date style. It also means that I have had to redefine the various date fields. The eventdate gives the recording date of a particular song or other portion of a recording, the origdate the recording date of an entire album, and the date the publishing date of that album. The earlier date is the one that will appear in citations and at the head of reference list entries. Please see the full documentation in section 5.1, above.

- The specification for **video** entries has also been clarified. For television series, the episode and series numbers go in booktitleaddon instead of titleaddon and, as with music entries, the eventdate will hold the original broadcast date of such an episode, or perhaps the recording/performance date of, e.g., an opera on DVD. The origdate will still hold the original release date of a film, and the date the publishing or copyright date of the medium you are referencing. The earlier date, once again, is the one that will appear in citations and at the head of reference list entries. Please see the full documentation in section 5.1, above.

- In **suppbook** entries, the *Manual* now requires you to provide the page range (in the pages field) for the specific part you are citing, e.g., an introduction, foreword, or afterword.

- The author-date style now prefers longer bibstrings in the list of references, bringing it into line with the notes & bibliography style. Generally, the package will take care of this for you, but if you've been using abbreviated strings in note fields, for example, you may want to change them so that they conform with the strings the package provides. In some circumstances the \partedit macro, and its relatives, may help. See section 5.3.1.

- When a descriptive phrase is used as an **author**, you can now omit an initial definite or indefinite article, which will help with alphabetization in the bibliography.

- A DOI is now preferred to a URL, if both are available.

- On the same subject, a revision date (or similar) is preferred to an access date for online material. You can use the new **userd** field to change the string introducing the urldate, which defaults to being an access date.

- Special imprints are now separated from their parent press by a forward slash rather than a comma, so can just be added to the publisher field with the usual keyword and.

- The 16th edition of the *Manual* is less than enthusiastic about the use of "Anon." as the author, preferring instead that the title or the journaltitle take its place. If you do decide to get rid of "Anon.," new facilities provided by Biber — see next entry — should mean that biblatex no longer requires assistance when alphabetizing such author-less entries.

- The Chicago-specific setting of the Biber-only command, \DeclareSortingScheme =cms, allows non-standard fields to be considered by biblatex's sorting algorithms, which should reduce the instances where you need a sortkey or the like in your entries.

- The Chicago-specific setting of the Biber-only command \DeclareLabelname allows biblatex to find a name ("label") for citations outside the standard name fields, including, notably, in the name[a-c] fields. This should reduce the instances where you need a shortauthor field to provide such a name.

Other New Features:

- For reprinted books, you can now present more detailed publishing information about the original edition using the new **origlocation** and **origpublisher** fields. You can also use the origlocation in letter or misc (with entrysubtype) entries to identify where a published or unpublished letter was written. These uses apply to both Chicago styles.

- Thanks to a patch sent by Kazuo Teramoto, you can now take advantage of biblatex's facilities for citing **eprint** resources. There is also a new eprint option, set to true by default, which controls the printing of this field in both Chicago styles. You can set the option both in the preamble and in the options field of individual entries. The field will always print in **online** entries.

- I have added a new citation command, \citejournal, to the notes & bibliography style to allow you to present journal articles using an alternative short note form, which may be a clearer form of reference in certain circumstances. Such short notes will present the name of the author, the journaltitle, and the volume number.

- I have included a very slightly modified version of the standard biblatex \citeauthor command, which may be useful for references to works from classical antiquity.

- I have added a new cmsdate=full switch to the author-date style, which only affects citations in the text, and means that a full date specification will appear there, rather than just the year. If you follow the *Manual's* recommendations concerning newspaper and magazine articles only appearing in running text and not in the reference list, this option will help.

- I have added a new avdate option to the author-date style, set to true by default in biblatex-chicago.sty. This alters the default setting of \DeclareLabelyear in **music**, **review**, and **video** entries to take account of specialized instructions in the *Manual* for finding dates to appear in citations and at the head of reference list entries. Setting avdate=false in the options when you load biblatex-chicago restores the default settings for all entry types. See avdate in section 5.4.2.

- The *Manual* has added recommendations for citing blogs, which generally will need an **article** entry with magazine entrysubtype. You can identify a blog as such by placing "blog" in the location field. If you want to cite a comment to a blog or to other online material, the **review** entry type, entrysubtype magazine will serve. The **eventdate** dates the comment, and any time stamp that is required can go in nameaddon. These instructions work in both specifications.

- Photographs are no longer presented differently from other sorts of artworks so, in effect, in both styles, the **image** type is now a clone of **artwork**, though retained for backward compatibility.

- Following a request by Kenneth Pearce, I have added new facilities for presenting **shorthands** in both Chicago styles. In both, there are two new bibenvironments which you can set using the env option to the \printshorthands command: losnotes formats the list of shorthands so that it can be presented in a footnote, while losendnotes does the same for endnotes. In both styles, there is a new preamble option, shorthandfull, which prints the full bibliographical information of each entry inside the list of shorthands, allowing such a list effectively to replace a bibliography or list of references. In the author-date style, you need to set the cmslos=false option as well, in order for this to work. In the notes & bibliography style, I have added a new citation command, \shorthandcite, which prints the shorthand even in the first citation of a given work.

- Following suggestions by Roger Hart, I have implemented three new field-exclusion options in the notes & bibliography style. In all three cases, the field in question will always appear in the bibliography, but not in long notes, which may help to save space. The fields at stake are addendum, note, and series, controlled respectively by the new addendum, notefield, and bookseries options. All of these are set to true using the new completenotes option in chicago-notes.cbx, but you can change the settings either in the preamble or in the options field of individual entries. Please see the documentation of these options in section 4.4.2, above, for details on which entry types are excluded from their scope.

- Thanks to a coding suggestion from Gildas Hamel, I have redefined the \bibnamedash in biblatex-chicago.sty, which should now by default look a little better in a wider variety of fonts.

- At the request of Baldur Kristinsson, I have added \DeclareLanguageMapping commands to biblatex-chicago.sty for all the languages biblatex-chicago currently provides. If you load the style in the standard way, you no longer need to provide these mappings manually yourself.

- I have improved the date handling in both styles, particularly with regard to date ranges.

0.9.8d: Released November 15, 2011

- Some minor fixes to both styles for compatibility with biblatex 1.7.

- Kenneth Pearce found an error in the formatting of bookinbook titles in the author-date style's list of shorthands. This should work properly now.

- Jonathan Robinson spotted some inconsistencies in the way the notes & bibliography style interacts with the hyperref package. Following his suggestion, short notes now point to long notes when the latter are available, but to bibliography entries instead when you have set the short option.

0.9.8c: Released October 12, 2011

- Emil Salim pointed out some rather basic errors in the presentation of inproceedings and proceedings entries, errors that have been present from the first release of the style(s). These should now, belatedly, have been put right.

- Minor improvements to coding and documentation.

0.9.8b: Released September 29, 2011

- Bad Dates: Christian Boesch alerted me to some date-formatting errors produced when using the styles with the german option to babel. A little further investigation revealed similar problems with french, and before long it became clear that date handling in biblatex-chicago was generally, and significantly, sub-optimal. The whole system should now be more robust and more accurate.

- The new date-handling code shouldn't require any changes to your .bib files, but users of the author-date style may want to have a look at the documentation of the letter and misc entry types, and of the four date fields, for some information about how the changes could simplify the creation of their databases.

- Various other minor improvements.

0.9.8a: Released September 21, 2011

- Fixed a series of unsightly errors in the author-date style, discovered while working on the pending update to the 16th edition.

- Fixed bugs uncovered in both the author-date and the notes & bibliography styles thanks to Charles Schaum's adventurous use of the origyear field.

- Added two new bibstrings to the cms-*.lbx files to fix potential bugs in some of the audiovisual entry types.

0.9.8: Released August 31, 2011

- Starting with biblatex version 1.5, in order to adhere to the author-date specification you will need to use Biber to process your .bib files, as BIBTEX (and its more recent variants) will no longer provide all the required features. Unfortunately, however, the current release of Biber (0.9.5) contains bugs that make it tricky to use with biblatex-chicago. These bugs have been addressed in 0.9.6 beta, which is available for various operating systems in the development subdirectory of your SourceForge mirror, e.g., UK mirror. (If, by the time you read this, Biber 0.9.6 has already been released, then so much the better.) Please see the start of cms-dates-sample.pdf for more details.

- The switch to Biber for the author-date specification means that biblatex now provides considerably enhanced handling of the various date fields. I have attempted to document the relevant changes in cms-dates-sample.pdf and in the **date** discussion in section 5.2, above, but in my testing the only alterations I've so far had to make to my .bib files involve adhering more closely to the instructions for specifying date ranges. Biber doesn't like {1968/75}, and will ignore it. Either use {1968/1975} or use {1968--75} in the year field.

- In the notes & bibliography style, and mainly in article, letter, misc, and review entries, previous releases of biblatex-chicago recommended using the \isdot macro when you needed both to define a field and not have it appear in the printed output. This mechanism no longer works in biblatex 1.6, and while addressing the problem I realized that relying on it covered over some inconsistencies and bugs in my code, so from this release forward you will need to modify your .bib and .tex files to use other, more standard mechanisms to achieve the same ends, in particular the \headlesscite commands and declaring useauthor=false in the options field. Please consult the documentation in section 4.3.1, s.v. "\isdot," for a list of example entries where you can see these changes at work.

Other New Features:

- Fixed the \smartcite citation command in, and added a \smartcites command to, chicago-notes.cbx, so that the notes & bibliography style no longer prints parentheses around citations produced using \autocite(s) commands inside \footnote commands. Many thanks to Louis-Dominique Dubeau for pointing out this error.

- Rembrandt Wolpert and Aaron Lambert pointed out an issue with a command (\lbx @fromlang) that biblatex no longer defines, and Charles Schaum very kindly suggested a temporary workaround in a newsgroup post, a workaround that should no longer be necessary.

- Version 1.6 of biblatex no longer allows you to redefine the minnames and maxnames options in the \printbibliography command, so I've defined minbibnames

and `maxbibnames` in biblatex-chicago.sty, instead. These parameters have been available since version 1.1, so this is now the earliest version of biblatex that will work with the Chicago styles. Of course, if the (Chicago-recommended) values of these options don't suit your needs, you can redefine them in your document preamble.

0.9.7a: Released March 17, 2011

- Added `\smartcite` command to chicago-notes.cbx so that the notes & bibliography style will work with biblatex 1.3.

- Added bibstrings `byconductor` and `cbyconductor` to the .lbx files, mistakenly omitted in version 0.9.7.

- Minor fixes to the docs.

0.9.7: Released February 15, 2011

Obsolete and Deprecated Features:

- The **customa** and **customb** entry types are now obsolete. Any such entries will be ignored. Please change any that remain to **letter** and **bookinbook**, respectively.

- If you still have any **customc** entries containing introductions, prefaces, or the like, please change them to **suppbook**. I have recycled customc for another purpose, on which see below.

Other New Features:

- At the request of Johan Nordstrom, I have added three new audiovisual entry types to both styles, **audio**, **music**, and **video**. The documentation of audio in sections 4.1 and 5.1 above contains an overview of the three, and the details for each type are to be found under their individual headings.

- I have transformed the **customc** entry type to enable alphabetized cross-references — the "c" is meant to be mnemonic — to other, separate entries in a reference list or bibliography. In particular, this facilitates cross-references to other names in a list, rather than to other works. In author-date, in a procedure recommended by the *Manual*, this now allows you to expand shorthands inside the reference list rather than in a list of shorthands. In both styles, you can now provide a pointer to the main entry if a reader is looking an author up under, e.g., a pseudonym or other alternative name.

- I have introduced the **userc** field, intended to simplify the printing of the cross-references provided by customc entries. The standard `\nocite` command works as well, but the additional mechanism may be more convenient in some circumstances.

- You can now provide an **eventdate** in music entries to identify, e.g., a particular recording session. It will be printed just after the title.

- In the notes & bibliography style, I have now implemented the **shorthandintro** field, which allows you to change the string introducing a shorthand in the first, long note. It works just as it does in the standard biblatex styles.

- At the request of Scot Becker, I have added six new field-exclusion options to both styles, all of which can be set both in the document preamble and/or in the options field of individual .bib entries. Three of these — `doi`, `isbn`, and `url` — are standard biblatex options, the others — `bookpages`, `includeall`, and `numbermonth` — are chicago-specific. See the docs in sections 4.4.2 and 5.4.2, above.

- At the request of Charles Schaum, I've added the `juniorcomma` option to both styles, which can be set in the document preamble and/or in the options field of individual entries. It allows you to get the traditional comma between a surname and "Jr." or "Sr."

- Fixed an old inaccuracy in the presentation of "Jr." and "Sr.," so that they now appear at the end of names printed surname first in bibliographies and reference lists.

- Thanks to Andrew Goldstone, I fixed some old inaccuracies in the syntax of shortened notes and bibliography entries presenting multiple contributions to one multi-author (or single-author) volume.

- I've altered the directory structure of the archive containing this release. Files were multiplying, and look set to multiply still further, so I've copied the structure used by Lehman for biblatex itself.

- Fixed an old bug, which I'd guess was triggered quite rarely, in the formatting of publication information in long notes.

- Fixed another bug in author-date where the colon separating titles and subtitles was in the wrong font. The biblatex punctfont option solved this.

- Fixed a punctuation bug in InReference entries in the notes & bibliography style. Also fixed title presentation in Reference entries in author-date.

- Fixed some inaccuracies in the tests establishing priority between date and origdate fields. These arose when date ranges were involved, and it's possible I haven't yet addressed all possible permutations of the problem.

- Added several new bibstrings to the cms-*.lbx files for the new audiovisual entry types. This means that the editortype fields can now be set to director, producer, or conductor, depending on your needs. You can also set the fields to none, which eliminates all identifying strings, and which is useful for identifying performers of various sorts.

- Minor improvements to documentation.

0.9.5a: Released September 7, 2010

- Quick fix for an elementary and show-stopping mistake in biblatex-chicago.sty, a mistake disguised if you load csquotes, which I do in all my test files. Mea culpa. Many thanks indeed to Israel Jacques and Emil Salim for pointing this out to me.

0.9.5: Released September 3, 2010

Obsolete and Deprecated Features:

- All the custom entry types — **customa**, **customb**, and **customc** — are now deprecated. They will still work for the time being, but please be aware that in the next major release they will no longer function, at least not as you might be expecting. Please change your .bib files to use **letter** (=**customa**), **bookinbook** (=**customb**), and **suppbook** (=**customc**) instead.

- If by some chance anyone is still using the old \custpunctc macro, it is now obsolete. It really shouldn't be needed, but let me know if I'm wrong.

Other New Features:

- The Chicago author-date style is now implemented in the package, and is fully documented in section 5, above.

- The default way of loading the style(s) has slightly changed. You should put either notes or authordate in the options to biblatex-chicago, e.g.:

      \usepackage[authordate,more options …]{biblatex-chicago}

- With the addition of the second Chicago style, I have thought it appropriate to alter both the name of the package and the names of the files it contains. The package is now biblatex-chicago instead of biblatex-chicago-notes-df, and the following files have been renamed:

- chicago-notes-df.cbx is now chicago-notes.cbx

- chicago-notes-df.bbx is now chicago-notes.bbx

- sample.tex is now cms-notes-sample.tex

- sample.pdf is now cms-notes-sample.pdf

- chicago-test.bib is now notes-test.bib

- biblatex-chicago-notes-df.pdf (this file) is now biblatex-chicago.pdf

The following files have been added:

- chicago-authordate.cbx

- chicago-authordate.bbx

- cms-dates-sample.tex

- cms-dates-sample.pdf

- dates-test.bib

The following files have retained their old names:

- cms-american.lbx

- cms-french.lbx

- cms-german.lbx

- cms-ngerman.lbx

- biblatex-chicago.sty

- I have implemented the pubstate field, slightly differently yet compatibly in the two styles, to provide a simpler mechanism for identifying a reprinted book. In the author-date style, it is highly recommended you use it, as it sorts out some complicated formatting questions automatically. In the notes & bibliography style it isn't strictly necessary, but may be useful anyway and easier to remember than the old system. See the documentation under pubstate in sections 4.2 and 5.2, above.

- Users of biblatex-chicago-notes no longer need a shortauthor field in author-less manual entries, or in author-less article or review entries with a magazine entry-subtype. The package will now automatically take an author for short notes from the organization field for manual entries and from the journaltitle field for the others. You can still use a shortauthor field if you want, but it's no longer necessary. (This also holds for chicago-authordate.)

- Date presentation in the misc entry type (with entrysubtype) has changed to fix an inaccuracy. You can now use the date and origdate fields to distinguish between two sorts of archival source: letters and "letter-like" sources use origdate, interviews and other non-letters use date. The only difference is in how the date is printed, so current .bib entries will continue to work fine, albeit with minor inaccuracies in the case of non-letter-like sources. See the docs on **misc** in sections 4.1 and 5.1, above.

- When only one date is presented in a patent entry — either in the date or origdate field — this will now always be used as the filing date. In biblatex-chicago-notes, this makes a change from the previous (incorrect) behavior.

- I have included the option dateabbrev=false in the default settings for biblatex-chicago-notes. This ensures that the long month names are printed, as otherwise recent releases of biblatex print the abbreviated ones by default.

- The provision of punctuation in entrysubtype classical entries has been improved, allowing the comma to appear before certain kinds of location specifiers even when citing works by their traditional divisions. See *Manual* 17.253. (This applies to both Chicago styles.)

- The number field in article, periodical, and review entries now allows you to include a series or range of numbers in the field, with the style automatically providing the correct bibstring (singular or plural).

- I have removed and altered bibstrings in the .lbx files to take advantage of the new \bibsstring and \biblstring commands in biblatex, and added one new string (origpubyear) needed by biblatex-chicago-authordate.

0.9a: Released March 20, 2010

- Quick fixes for compatibility with biblatex 0.9a.

0.9: Released March 18, 2010

Obsolete and Deprecated Features:

- The **userd** field is now obsolete. All information it used to hold should be placed in the edition field.

- The **origyear** field is now obsolete in biblatex. It has been replaced by **origdate**, and because the latter allows a full date specification, I have been able to make the operation of customa (= letter), misc (with an entrysubtype), and patent entries more intuitive. The RELEASE file contained in this package gives the short instructions on how to update your .bib files, and you can also consult the documentation of those entry types above.

- The modified csquotes.cfg file I provided in earlier releases is now obsolete, and has been removed from the package. Please upgrade to the latest version of csquotes and, if you are still using my modified .cfg file, remove it from your TEX search path, or at the very least excise the code I provided.

Other New Features:

- Added the files cms-german.lbx (with its clone cms-ngerman.lbx) and cms-french .lbx, which allow the creation of Chicago-like references in those languages. See section 7 above for details on usage.

- Added the annotation package option to allow the creation of annotated bibliographies. This code is still not entirely polished yet, but it is usable. Please see page 28 above for instructions and hints.

- Added biblatex's new **bookinbook** entry type, which currently functions as an alias of the customb type. As biblatex now provides standard equivalents for all of the custom types I initially found it necessary to provide — letter = customa, bookinbook = customb, and suppbook & suppcollection = customc — it may soon be time to prune out the custom types to enhance compatibility with other biblatex styles. I shall give plenty of warning before I do so.

- In line with the new system adopted in biblatex 0.9, using the editortype field turns off the usual string concatenation mechanisms of the Chicago style. See Lehman's RELEASE file for a discussion of this.

- I have added support for the new editor[a–c] and editor[a–c]type fields, and they work just as in standard biblatex, though I'm uncertain how much use they'll get from users of the Chicago style.

- I have added many bibstrings to the .lbx files to help with internationalization. The new ones that you might want to use in your .bib files include: pseudonym, nodate, revisededition, numbers, and reviewof. Please see section 7 for a fuller list.

0.8.9d: Released February 17, 2010

- Chris Sparks and Aaron Lambert both found formatting bugs in the 0.8.9c code. I've fixed these bugs, and am releasing this version now, the last in the 0.8.9 series. The next release of biblatex-chicago-notes-df, due as soon as possible, will contain many more significant changes, including those necessary for it to function properly with the recently-released biblatex version 0.9. In the meantime, at least version 0.8.9d should produce more accurate output.

0.8.9c: Released November 4, 2009

- Emil Salim noticed that the *ibidem* mechanism wasn't working properly, printing the page number after "Ibid" even when the page reference of the preceding citation was identical. The fix for this involved setting loccittracker=constrict in biblatex-chicago.sty, something you'll have to do manually yourself if you're loading the package via a call to biblatex rather than to biblatex-chicago.

- Several users have reported unwanted behavior when repeated names in bibliographies are replaced with the bibnamedash. This release should fix both when the bibnamedash appears and what punctuation follows it.

0.8.9b: Released September 9, 2009

- Fixed a long-standing bug in formatting names in the bibliography. The package now correctly places a comma after the reversed name that begins the entry, using biblatex's \revsdnamedelim command. Many thanks to Johanna Pink for catching my rather egregious error.

- While fixing some formatting errors that cropped up when using the newest version of biblatex (0.8h at time of writing), I also spotted some more venerable bugs in the code for using shortened cross-references for citing multiple entries in a collection of essays or letters. I believe this now works correctly, but please let me know if you discover differently.

- Joseph Reagle noticed that endnote marks (produced using the endnotes package) did not receive the same treatment as footnote marks. I have rectified this, placing the code in biblatex-chicago.sty so that you can turn it off either by using the old package-loading system or by setting the footmarkoff package option when loading biblatex-chicago.

- Updates to Lehman's csquotes package have rendered my modifications in csquotes.cfg obsolete. Please use the latest version of csquotes (4.4a at time of writing) and ignore my file, which will disappear in a later release.

- At the request of Will Small, I have included some code, still in an alpha state, to allow you to specify, in the bibliography, the original publication details of essays which you are citing from later reprints (a *Collected Essays* volume, for example). See the documentation above under the reprinttitle field if you would like to test this functionality.

0.8.9a: Released July 5, 2009

- Slight changes for compatibility with biblatex 0.8e. The package still works with 0.8c and 0.8d, as well.

0.8.9: Released July 2, 2009

Obsolete and Deprecated Features:

- The **single-letter bibstrings** (\bibstring{a}, \bibstring{b}, etc.) are now obsolete. You should replace any still present in your .bib file with \autocap commands — see § 3.8.4 of biblatex.pdf.

Other New Features:

- The default way of loading the package is now with

  \usepackage[further-options]{biblatex-chicago}

  rather than

  \usepackage[style=chicago-notes-df,further-options]{biblatex}.

  Please see section 4.5.1 above for details and hints.

- Package-specific bibstrings have been removed from the .cbx and .bbx files and are now gathered in a new file, **cms-american.lbx**, which changes the way the package interacts with **babel**. It is now somewhat simpler if you want the defaults, but somewhat more complex if you require non-standard features. Please see section 9 above for more details.

- Two new entry types have been added: **artwork** for works of visual art excluding photographs, and **image** for photographs. See the documentation of artwork for how to create .bib entries for both types.

- Added the new bibliography and entry option **usecompiler**, set to true by default. This streamlines the code that finds a name to head an entry (**author -> editor [or namea] -> translator [or nameb] -> compiler [namec] -> title**). The whole system should work more consistently now, but do see the author and namec documentation for improved notes on how to use it.

- Added the new bibliography option **footmarkoff**, to turn off the optional in-line (as opposed to superscript) formatting of the marks in foot- or endnotes. You only need this if you load the package with the new default \usepackage{biblatex-chicago}; users loading it the old way get default LaTeX formatting.

- At Matthew Lundin's request, I have added the citation command **\headlesscite**, which works like \headlessfullcite but allows biblatex to decide whether to print the full or the short note.

- Fully adopted biblatex's system for providing end-of-entry punctuation, which will solve some of the bugs users have been finding. See section 4.5.2, above, and do please let me know if inconsistencies remain.

- Added a modified **csquotes.cfg** file to address issues users were having when using the XeLaTeX engine in combination with biblatex-chicago. See section 9, above.

- Added natbib option to allow users of the default setup to continue to benefit from biblatex's natbib compatibility code. Thanks to Bennett Helm for pointing out this issue.

- Added a **shorthandibid** option to allow the printing of *ibid.* in consecutive references to an entry that contains a shorthand field. Thanks to Chris Sparks for calling my attention to this problem.

- While investigating the preceding, I noticed failures when combining the short option with a shorthand field. The package now actually does what it has always claimed to do under **shorthand**.

- Many small bug fixes and improvements to the documentation.

To Do:

- The shorthand vs *ibid.* question may need more careful addressing in some cross references, and also in relation to the noibid package option.

- Charles Schaum has quite rightly pointed out the inconsistency in my naming conventions — biblatex-chicago.sty as opposed to chicago-notes-df.cbx, for example. I'm going to delay a decision on which way to go with this until a later release.

0.8.5a: Released June 14, 2009

- Quick and dirty fixes to bibliography strings to allow compatibility with biblatex version 0.8d. If you are still using 0.8c, then I would wait for the next version of biblatex-chicago-notes-df, which is due soon. See README.

0.8.5: Released January 10, 2009

Obsolete and Deprecated Features:

- The **\custpunct** commands are now deprecated — Lehman's "American" punctuation tracking facilities should handle quoted text automatically, assuming you remember always to use **\mkbibquote** in your database. If you still need \custpunct, please let me know, as it may be an error in the style.

- With \custpunct no longer needed, the toggles activated by placing "plain" in the **type** or **userb** fields are also deprecated.

Other New Features:

- At least **biblatex 0.8b** is now required — 0.8c works fine, as well.

- I now *strongly recommend* that you use **babel** with "american" as the main text language. See section 9 above for further details.

- The **customc** entry type has been revised, allowing you to cite any sort of supplementary material using the **type** field instead of relying on toggles in the introduction, afterword, and foreword fields, though these latter still work. The two new entry types **suppbook** and **suppcollection** are both aliased to customc, and therefore work in exactly the same way.

- The new entry type **suppperiodical** is aliased to **review**.

- The new entry type **letter** is aliased to **customa**.

- In **inreference** entries the postnote field of all \cite commands is now treated like data in lista, that is, it will be placed within quotation marks and prefaced with the appropriate string. The only difference is that you can only put one such article name in postnote, as it isn't a list field.

- I've set the new biblatex option usetranslator to true by default, which means entries will automatically be alphabetized by their translator in the absence of an author or an editor.

- A host of small formatting errors were eliminated, nearly all of them through adopting Lehman's punctuation tracker.

- In the main body of this documentation, I've added some **color coding** to help you more quickly to identify entry types and fields that are either new or that have undergone significant revision.

To Do:

- Separate out "options" from the basic citation "style," using a LaTeX style file. This is an architectural change recommended by Lehman.

0.8.2.2: Released November 24, 2008

- Fixed spurious commas appearing in some bibliography entries, spotted by Nick Andrewes. While investigating this I noticed a more general problem with punctuation after italicized titles ending with question marks or exclamation points. This will be addressed in forthcoming revisions both of biblatex and of this package.

- Nick also reported some problems with spurious punctuation in the bibliography when using XeLaTeX. I haven't yet been able to pin down the exact cause of these, but if you are using XeLaTeX and are having (or have solved) similar problems I'd be interested to hear from you.

0.8.2: Released November 3, 2008

- Fixed several formatting glitches between citations in multicite commands (spotted by Joseph Reagle) and also after some prenotes.

0.8.1: Released October 22, 2008

Obsolete and Deprecated Features:

- The **origlocation** field is now obsolete, and has been replaced by **lista**. Please update your .bib files accordingly.

- The single-letter **\bibstring** commands I provided in version 0.7 are now deprecated. In most cases, you'll be able to take advantage of the automatic contextual capitalization facilities introduced in this release, but if you still need the single-letter \bibstring functionality then you should switch to \autocap, as I shall be removing the single-letter bibstrings in a future release. See above under **\autocap** for all the details.

- The **userd** field is now deprecated, as biblatex 0.8 allows all forms of data to be included in the edition field. I shall be removing userd in a future release, so please update your .bib files as soon as is convenient.

Other New Features:

- Updated the .bbx and .cbx files to work with biblatex 0.8. This most recent version of biblatex is now required for biblatex-chicago-notes-df to work.

- Added the **usera** field, which holds supplemental information about a journaltitle in article and review entries. See the documentation of the field for details.

- Added the **\citetitles** multicite command to fix a problem with spurious punctuation when multiple titles were listed.

- Added the **\Citetitle** command to help with automatic capitalization of titles when they occur at the beginning of a note.

- Minor punctuation fixes in biblatex-chicago-notes-df.bbx.

To Do:

- Integrate biblatex's American punctuation facilities.

- Separate out "options" from the basic citation "style," using a LaTeX style file. This is an architectural change recommended by Lehman.

- Investigate and possibly integrate the new entry types provided in biblatex 0.8.

0.7: First public release, September 18, 2008