

# bath-bst: Harvard referencing style as recommended by the University of Bath Library

Maintainer: Alex Ball\*

Package v7.0 – 1 April 2025

## 1 Introduction

This package provides a **BibTeX** style to format reference lists in the **Harvard style** recommended by the University of Bath Library. It should be used in conjunction with **natbib** for citations.

### 1.1 Installation

The files you need are included in the zip archive available from **GitHub**. If you use the zip archive from **CTAN**, you will need to run `luatex bath-bst.dtx` to generate them.

You can use this style simply by copying the `bath.bst/bathx.bst` files into your working directory, that is, the directory holding the main `.tex` file for your document. If you want the style to be available for all your documents without having to copy it over each time, you can install it using the instructions below.

#### 1.1.1 Managed way

The latest stable release of `bath-bst` has been packaged for TeX Live and MiKTeX. If you are running TeX Live and have `tlmgr` installed, you can install the package simply by running `tlmgr install bath-bst`. If you are running MiKTeX, you can install the package by running `mpm --install=bath-bst`. Both `tlmgr` and `mpm` have GUI versions that you might find friendlier.

#### 1.1.2 Automated way

A makefile is provided which you can use with the Make utility on UNIX-like systems:

- Running `make source` generates the derived files
  - `README.md`
  - `bath.bst` and `bathx.bst`
  - `bath-bst-v1.tex`
  - `bath-bst.bib` and `bath-bst-v1.bib`
  - `bath-bst.ins`
- Running `make` generates the above files and also `bath-bst.pdf` and `bath-bst-v1.pdf`. Ensure you have (at least) the `luatexja`, `adobemapping` and `ipaex` packages installed first.
- Running `make inst` installs the files in the user's TeX tree. You can undo this with `make uninst`.
- Running `make install` installs the files in the local TeX tree. You can undo this with `make uninstall`.

---

\*To contact the maintainer about this package, please visit the repository where the code is hosted: <https://github.com/alex-ball/bathbib>.

### 1.1.3 Manual way

You do not need to follow the first step if you downloaded the zip archive from [GitHub](#). You do not need to follow the second step if you already have the PDF documentation.

1. Run `luatex bath-bst.dtx` to generate the source files.
2. Compile `bath-bst.dtx` with [LuaLaTeX](#) and BibTeX to generate the documentation. You will need, among other things, the [luatexja](#), [adobemapping](#), [ipaex](#) and [haranoaji](#) packages installed; this is just for the documentation, not for the BibTeX style itself. To generate the version 1 tests and documentation, compile `bath-bst-v1.tex` with LuaLaTeX and BibTeX.
3. If you are using TeX Live, find your home TeX tree using the following command at the command prompt/terminal:

```
kpsewhich -var-value=TEXMFHOME
```

If you are using MikTeX, consult the MikTeX manual entry for [integrating local additions](#). You can use one of the roots (TeX trees) already defined – preferably one of the User roots – or set up a new one.

4. Move the files to your TeX tree as follows:
  - `source/bibtex/bath-bst: bath-bst.dtx, (bath-bst.ins)`
  - `bibtex/bst/bath-bst: bath.bst, bathx.bst`
  - `doc/bibtex/bath-bst: bath-bst.pdf, bath-bst-v1.pdf, README.md`
5. You may then have to update your installation’s file name database before TeX and friends can see the files.

## 2 Choosing which variant to use

The data model offered by the standard BibTeX styles, and even the extended `natbib` variants, is not really rich enough to support the nuances of the Harvard (Bath) style. This means design decisions have to be made about whether to attempt some level of compatibility with other styles or craft something utterly unique.

In the first version of `bath-bst`, the intention was to set things up so that, if the same `.bib` file was used with a different style, the features peculiar to the Harvard (Bath) style would be ignored and the remaining information would come out in a sensible arrangement. The recommendations of the style’s documentation were to use the standard entry types as much as possible (though more semantic aliases were provided), and minimal new fields were introduced. For the more exotic (in BibTeX terms) demands of the style, extensive use was made of the (standard) `note` and (non-standard) `titleaddon` fields to place information properly. If you have written a `.bib` file according to the principles in that first version, the `bath` style will still work as advertised for you.

The second version introduced a new variant, `bathx` (‘Bath extended’), which has a different aim. The idea with this one is that `.bib` files written for this style will be rendered just the same by the companion `biblatex-bath` style. Where possible, features from the latter have been emulated for BibTeX; otherwise, the ‘cheats’ used in this style will also work under `biblatex`.

Happily, it has been possible to do this while keeping most of the code in common. The main differences between two versions are as follows:

- In `bath`, the `titleaddon` field is printed bare, while in `bathx` it is wrapped in square brackets.
- In `bathx` online items are marked with ‘[Online]’ automatically, while in `bath` you have to mark them thus yourself.
- In `bathx` undated items are marked with ‘n.d.’ automatically, while in `bath` you have to give ‘n.d.’ as the value of `year` yourself.

One other change worth noting regards URL access dates. In version 1, the advice was to put them in `urldate`. From version 2, the advice is to put them, perhaps counter-intuitively, in `urlyear`. The reason is that `biblatex` will complain if the value of `urldate` is in the wrong format, but will happily print the contents of `urlyear` literally. If you don’t care about `biblatex` compatibility, you can continue to use `urldate`.

### 3 Using the style

To use the style, add these lines to your preamble:

```
\usepackage{natbib}
\setlength{\bibhang}{0pt}
\newcommand*{\urlprefix}{Available from: }
\newcommand*{\urldateprefix}{Accessed }
\bibliographystyle{bathx}
```

Remember also to specify your `.bib` file at the end of the document:

```
\bibliography{file}
```

To make a citation in the text, use the key that corresponds to the entry in your `.bib` file:

While collections can be supplemented by other means `\citep{devlin.etal2021lipp}`, the absence of an invisible collection amongst historians is noted by `\citet[p.556]{stieg1981inh}`. It may be, as `\citet{burchard1965hhl}` points out, that they have no assistants or are reluctant to delegate, or it may be down to economic factors `\citep{adams2009tc1, adams2014tc2, gb.pa2014, adams2017tc3}\dots`

While collections can be supplemented by other means (Devlin, Martin and Ostrovnaya, 2021), the absence of an invisible collection amongst historians is noted by Stieg (1981b, p.556). It may be, as Burchard (1965) points out, that they have no assistants or are reluctant to delegate, or it may be down to economic factors (Adams, 2009, 2014; *Pensions Act*, 2014; Adams, 2017)...

Please refer to the documentation for `natbib` for the full range of commands available for in-text citations. Be aware that the `natbib` option `sort` will sort citations in bibliography order, rather than the chronological order demanded by Harvard (Bath) – so don't use it!

### 4 Breaking changes between versions

The following sections list implementation changes that might affect the way in which you write your bibliography database files.

In addition to the changes listed, when upgrading between major versions, you should expect some changes in formatting relating to the evolution of the Harvard (Bath) style itself. Some of the style's internal settings and structures may also change without warning.

#### 4.1 Version 3

- Journal titles are now coerced to sentence case, so any capital letters you want to keep need to be protected with braces.
- Entries of type `unpublished` are now marked as unpublished. If you have been using this for an item that should not be marked as unpublished (e.g. archive photographs, law reports), you can use `booklet` instead: this is the other entry type (along with `misc`) that has a `howpublished` field instead of a `publisher`.
- Some changes to the Harvard (Bath) style cannot be applied automatically (e.g. the update to how standards are referenced), so you may need to update your `.bib` file accordingly.



## 4.2 Version 5

- In the 2021 revision of the style, the titles of broadcast media (TV, radio) became upright, with series information for titled episodes being italic. To get this new formatting, you must use the `type` field as shown.
- Series names (except those in `legislation` and `jurisdiction` entries) are now coerced to sentence case, so any capital letters you want to keep need to be protected with braces.
- The ‘[Online]’ tag is now also triggered by `urldate/urlyear`. In cases where it occurs directly after the title, it is no longer left behind (after the year) if the title is moved to the head of the reference.
- The `nameaddon` field is now supported for annotating author lists with, say, a social media handle.
- The `article` and `manual` entry types now support the `month` field.
- Maps need to be given the `book` entry type; previously `manual` was recommended.

## 4.3 Version 6

- Entry type clarifications such as ‘Photograph’ and ‘Computer program’ are now handled differently from translated titles, in particular as regards their position relative to the version, and how they interact with the ‘[Online]’ tag. As a result, they should now be placed in `entrysubtype` rather than `titleaddon`.
- All elements of the title block (including `titleaddon`, `version`, and `entrysubtype`) now move to the head of the reference in the absence of an author/editor.


# 5 Examples


The examples below are shown in three parts. The first, marked with , shows an extract from the *Referencing guide: Harvard Bath* or *Referencing images*. The second, marked with , shows the reference as formatted by Bib<sub>T</sub>EX. The last shows how the reference was entered in the `.bib` file. The bottom right corner shows the source of the example: ‘RX’ indicates the ‘Reference examples (A-Z)’ section of the Guide; ‘RL’ indicates the ‘Organise a reference list’ section of the Guide; ‘RI’ indicates *Referencing images*.

Some examples are highlighted in orange. This indicates that some fields have been ‘abused’ to achieve the right effect; in other words, they contain information that does not conform with their intended use. Some others make use of the `note` and `titleaddon` fields to achieve the right effect, where other styles might need the information placed differently. Particular care should be taken with such items when switching between different styles, though of course any item might need adjustment to take account of differing conventions.

## 5.1 Books and book chapters

### Book with author(s)

 Rang, H.P., Dale, M.M., Ritter, J.M., Flower, R.J. and Henderson, G., 2012. *Rang and Dale’s pharmacology*. 7th ed. Edinburgh: Elsevier Churchill Livingstone.

 Rang, H.P., Dale, M.M., Ritter, J.M., Flower, R.J. and Henderson, G., 2012. *Rang and Dale’s pharmacology*. 7th ed. Edinburgh: Elsevier Churchill Livingstone.

```
@book{rang.etal2012rdp,  
  author = {Rang, H. P. and Dale, M. M. and Ritter, J. M. and Flower, R. J. and Henderson, G.},  
  year = {2012},  
  title = {Rang and {Dale's} Pharmacology},  
  edition = {7},  
  address = {Edinburgh},  
  publisher = {Elsevier Churchill Livingstone}}
```

RX

- 📖 Open University, 1972. *Electricity and magnetism*. Bletchley: Open University Press.
- ⚙️ Open University, 1972. *Electricity and magnetism*. Bletchley: Open University Press.

```
@book{ou1972em,
  author = {Open University},
  year = {1972},
  title = {Electricity and Magnetism},
  address = {Bletchley},
  publisher = {Open University Press}}
```

RX

- 📖 Solomon, M.R., Askegaard, S., Hogg, M. and Bamossy, G.J., 2019. *Consumer behaviour: a European perspective*. 7th ed. Harlow: Pearson.
- ⚙️ Solomon, M.R., Askegaard, S., Hogg, M. and Bamossy, G.J., 2019. *Consumer behaviour: a European perspective*. 7th ed. Harlow: Pearson.

```
@book{solomon.etal2019cbe,
  author = {Solomon, M. R. and Askegaard, S. and Hogg, M. and Bamossy, G. J.},
  year = {2019},
  title = {Consumer Behaviour: a {European} Perspective},
  edition = {7},
  address = {Harlow},
  publisher = {Pearson}}
```

RX

### Book with editor(s) instead of author(s)

- 📖 Rothman, K.J., Greenland, S. and Lash, T.L., eds, 2008. *Modern epidemiology*. 3rd ed. Philadelphia, Pa.: Lippincott Williams & Wilkins.
- ⚙️ Rothman, K.J., Greenland, S. and Lash, T.L., eds, 2008. *Modern epidemiology*. 3rd ed. Philadelphia, Pa.: Lippincott Williams & Wilkins.

```
@book{rothman.etal2008me,
  editor = {Kenneth J. Rothman and Sander Greenland and Timothy L. Lash},
  year = {2008},
  title = {Modern Epidemiology},
  edition = {3},
  address = {Philadelphia, Pa.},
  publisher = {Lippincott Williams \& Wilkins}}
```

RX

- 📖 Internally, collection is an alias for book.

### Electronic book

- 📖 Haynes, W.M., ed., 2014. *CRC handbook of chemistry and physics* [Online]. 94th ed. Boca Raton, Fla.: CRC Press/Taylor and Francis. Available from: <http://www.hbcnetbase.com> [Accessed 16 June 2016].
- ⚙️ Haynes, W.M., ed., 2014. *CRC handbook of chemistry and physics* [Online]. 94th ed. Boca Raton, Fla.: CRC Press/Taylor and Francis. Available from: <http://www.hbcnetbase.com> [Accessed 16 June 2016].

```
@book{haynes2014crc,
  editor = {Haynes, W. M.},
  year = {2014},
  title = {{CRC} Handbook of Chemistry and Physics},
  edition = {94},
  address = {Boca Raton, Fla.},
  publisher = {CRC Press/Taylor and Francis},
  url = {http://www.hbcnetbase.com},
  urlyear = {16 June 2016}}
```

RX

📖 Blockley, D., 2021. *Engineering: a very short introduction* [Online]. Oxford: Oxford University Press. Available from: <https://www.amazon.co.uk/kindle> [Accessed 30 June 2021].

⚙️ Blockley, D., 2021. *Engineering: a very short introduction* [Online]. Oxford: Oxford University Press. Available from: <https://www.amazon.co.uk/kindle> [Accessed 30 June 2021].

```
@book{blockley2021evs,  
  author = {Blockley, D.},  
  year = {2021},  
  title = {Engineering: a Very Short Introduction},  
  address = {Oxford},  
  publisher = {Oxford University Press},  
  url = {https://www.amazon.co.uk/kindle},  
  urlyear = {30 June 2021}}
```

RX

### Book known by its title

❗ Unfortunately, to avoid breaking other use cases for the **book** entry type, to achieve the following format you must use the non-standard **reference** entry type.

📖 British National Formulary, 2020. 79th ed. London: Pharmaceutical Press.

⚙️ British National Formulary, 2020b. 79th ed. London: Pharmaceutical Press.

```
@reference{bnf2020,  
  year = {2020},  
  title = {{British National Formulary}},  
  edition = {79},  
  address = {London},  
  publisher = {Pharmaceutical Press}}
```

RX

❗ Similarly, to achieve the following formats you must use the non-standard **inreference** entry type.

📖 British National Formulary, 2020. 79th ed. *Aspirin*. London: Pharmaceutical Press, pp.280–281.

⚙️ British National Formulary, 2020a. 79th ed. *Aspirin*. London: Pharmaceutical Press, pp.280–281.

```
@inreference{aspirin2020bnf,  
  title = {Aspirin},  
  year = {2020},  
  booktitle = {{British National Formulary}},  
  edition = {79},  
  address = {London},  
  publisher = {Pharmaceutical Press},  
  pages = {280–281}}
```

RX

📖 British National Formulary, 2019. *Aspirin* [Online]. London: Pharmaceutical Press. Available from: [https://www.medicinescomplete.com/#/content/bnf/\\_456850132](https://www.medicinescomplete.com/#/content/bnf/_456850132) [Accessed 26 November 2019].

⚙️ British National Formulary, 2019. *Aspirin* [Online]. London: Pharmaceutical Press. Available from: [https://www.medicinescomplete.com/#/content/bnf/\\_456850132](https://www.medicinescomplete.com/#/content/bnf/_456850132) [Accessed 26 November 2019].

```
@inreference{aspirin2019bnf,  
  title = {Aspirin},  
  year = {2019},  
  booktitle = {{British National Formulary}},  
  address = {London},  
  publisher = {Pharmaceutical Press},  
  url = {https://www.medicinescomplete.com/#/content/bnf/_456850132},  
  urlyear = {26 November 2019}}
```

RX

❗ If you are using the same .bib file with BibTeX and biblatex, one unfortunate difference is that you must escape the hash symbol inside URLs with BibTeX but you must not escape it inside a URL with biblatex.

❗ Internally, `inreference` is an alias for `incollection`. Use `inreference` if you need compatibility with `biblatex-bath`.

❗ The style ensures the `booktitle` is used in title citation commands.

```
\cite{aspirin2019bnf}, \citet{bnf2020}, \citet{aspirin2020bnf}.
```

British National Formulary (2019), British National Formulary (2020b), British National Formulary (2020a).

## Chapter/paper from a collection (by different authors) in an edited book

📖 Burchard, J.E., 1965. How humanists use a library. In: C.F.J. Overhage and J.R. Harman, eds. *Intrex: report on a planning conference and information transfer experiments*. Cambridge, Mass.: MIT Press, pp.41–87.

⚙️ Burchard, J.E., 1965. How humanists use a library. In: C.F.J. Overhage and J.R. Harman, eds. *Intrex: report on a planning conference and information transfer experiments*. Cambridge, Mass.: MIT Press, pp.41–87.

```
@incollection{burchard1965hhl,  
  author = {Burchard, J. E.},  
  year = {1965},  
  title = {How Humanists use a Library},  
  editor = {C. F. J. Overhage and J. R. Harman},  
  booktitle = {Intrex: report on a planning conference and information transfer experiments},  
  address = {Cambridge, Mass.},  
  publisher = {MIT Press},  
  pages = {41-87}}
```

RL

- 📖 Reid, D.R., 1967. Physical testing of polymer films. In: S.H. Pinner, ed. *Modern packaging films*. London: Butterworths, pp.143–183.
- ⚙️ Reid, D.R., 1967. Physical testing of polymer films. In: S.H. Pinner, ed. *Modern packaging films*. London: Butterworths, pp.143–183.

```
@incollection{reid1967ptp,
  author = {D. R. Reid},
  year = {1967},
  title = {Physical Testing of Polymer Films},
  editor = {S. H. Pinner},
  booktitle = {Modern Packaging Films},
  address = {London},
  publisher = {Butterworths},
  pages = {143-183}}
```

RX

## 5.2 Articles and periodicals

### Journal article

- 📖 Stieg, M.F., 1981a. Continuing education and the reference librarian in the academic and research library. *Library journal*, 105(22), pp.2547–2551.
- ⚙️ Stieg, M.F., 1981a. Continuing education and the reference librarian in the academic and research library. *Library journal*, 105(22), pp.2547–2551.

```
@article{stieg1981cer,
  author = {Stieg, M. F.},
  year = {1981},
  title = {Continuing Education and the Reference Librarian in the Academic and Research Library},
  journal = {Library Journal},
  volume = {105},
  number = {22},
  pages = {2547-2551}}
```

RL

- 📖 Stieg, M.F., 1981b. The information needs of historians. *College and research libraries*, 42(6), pp.549–560.
- ⚙️ Stieg, M.F., 1981b. The information needs of historians. *College and research libraries*, 42(6), pp.549–560.

```
@article{stieg1981inh,
  author = {Stieg, M. F.},
  year = {1981},
  title = {The Information Needs of Historians},
  journal = {College and Research Libraries},
  volume = {42},
  number = {6},
  pages = {549-560}}
```

RL

- 📖 Newman, R., 2010. Malaria control beyond 2010. *British medical journal*, 341(7765), pp.157–208.
- ⚙️ Newman, R., 2010. Malaria control beyond 2010. *British medical journal*, 341(7765), pp.157–208.

```
@article{newman2010mcb,
  author = {Newman, R.},
  year = {2010},
  title = {Malaria control beyond 2010},
  journal = {{British} Medical Journal},
  volume = {341},
  number = {7765},
  pages = {157-208}}
```

RX



- 📄 Wetzstein, G., Ozcan, A., Gigan, S., Fan, S., Englund, D., Soljacic, M., Denz, C., Miller, D.A.B. and Psaltis, D., 2020. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588(7836), pp.39–47.
- ⚙️ Wetzstein, G., Ozcan, A., Gigan, S., Fan, S., Englund, D., Soljacic, M., Denz, C., Miller, D.A.B. and Psaltis, D., 2020. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588(7836), pp.39–47.

```
@article{wetzstein.etal2020iai,
  author = {Wetzstein, G. and Ozcan, A. and Gigan, S. and Fan, S. and
    Englund, D. and Soljacic, M. and Denz, C. and Miller, D. A. B. and
    Psaltis, D.},
  year = {2020},
  title = {Inference in Artificial Intelligence with Deep Optics and Photonics},
  journal = {Nature},
  volume = {588},
  number = {7836},
  pages = {39-47}}
```

RX

### Electronic journal article

- 📄 Devlin, S.M., Martin, A. and Ostrovnaya, I., 2021. Identifying prognostic pairwise relationships among bacterial species in microbiome studies. *PLOS computational biology* [Online], 17(11). Available from: <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501> [Accessed 9 December 2021].
- ⚙️ Devlin, S.M., Martin, A. and Ostrovnaya, I., 2021. Identifying prognostic pairwise relationships among bacterial species in microbiome studies. *PLOS computational biology* [Online], 17(11). Available from: <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501> [Accessed 9 December 2021].

```
@article{devlin.etal2021ipp,
  author = {Devlin, S. M. and Martin, A. and Ostrovnaya, I.},
  year = {2021},
  title = {Identifying Prognostic Pairwise Relationships Among Bacterial
    Species in Microbiome Studies},
  journal = {{PLOS} Computational Biology},
  volume = {17},
  number = {11},
  url = {https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501},
  urlyear = {9 December 2021}}
```

RX

- 📄 Steward, S., Connelly, D. and Robinson, J., 2020. Everything you should know about the coronavirus outbreak. *The pharmaceutical journal* [Online]. Available from: <https://www.pharmaceutical-journal.com/news-and-analysis/features/everything-you-should-know-about-the-coronavirus-outbreak/20207629.article> [Accessed 30 April 2020].
- ⚙️ Steward, S., Connelly, D. and Robinson, J., 2020. Everything you should know about the coronavirus outbreak. *The pharmaceutical journal* [Online]. Available from: <https://www.pharmaceutical-journal.com/news-and-analysis/features/everything-you-should-know-about-the-coronavirus-outbreak/20207629.article> [Accessed 30 April 2020].

```
@article{steward.etal2020eys,
  author = {Steward, S. and Connelly, D. and Robinson, J.},
  year = {2020},
  title = {Everything You Should Know About the Coronavirus Outbreak},
  journal = {The Pharmaceutical Journal},
  url = {https://www.pharmaceutical-journal.com/news-and-analysis/features/everything-you-should-know-about-the-coronavirus-outbreak/20207629.article},
  urlyear = {30 April 2020}}
```

RX

💡 If citing an article yet to be officially published, use the `pubstate` field (from `bibtex`) with the

keyword **inpress** for ‘in press’ and **inpreparation** or **submitted** (whichever is more accurate) for ‘preprint’.

- 📄 Liontou, C., Kontopodis, E., Oikonomidis, N., Maniotis, C., Tassopoulos, A., Tsiafoutis, I., Lazaris, E. and Koutouzis, M., 2019. Distal radial access: a review article. *Cardiovascular revascularization medicine* [Online], in press. Available from: <https://www.sciencedirect.com/science/article/pii/S1553838919303367> [Accessed 19 June 2019].
- ⚙️ Liontou, C., Kontopodis, E., Oikonomidis, N., Maniotis, C., Tassopoulos, A., Tsiafoutis, I., Lazaris, E. and Koutouzis, M., 2019. Distal radial access: a review article. *Cardiovascular revascularization medicine* [Online], in press. Available from: <https://www.sciencedirect.com/science/article/pii/S1553838919303367> [Accessed 19 June 2019].

```
@article{liontou.etal2019dra,  
  author = {Liontou, C. and Kontopodis, E. and Oikonomidis, N. and Maniotis, C. and  
    Tassopoulos, A. and Tsiafoutis, I. and Lazaris, E. and Koutouzis, M.},  
  year = {2019},  
  title = {Distal Radial Access: a Review Article},  
  journal = {Cardiovascular Revascularization Medicine},  
  pubstate = {inpress},  
  url = {https://www.sciencedirect.com/science/article/pii/S1553838919303367},  
  urlyear = {19 June 2019}}
```

RX

### Preprint or trial study in a digital repository

- 📄 Shah, I. and Corrick, I., 2016. *How should central banks respond to non-neutral inflation expectations?* Bath: University of Bath. *OPUS* [Online]. Available from: <http://opus.bath.ac.uk> [Accessed 4 May 2016].
- ⚙️ Shah, I. and Corrick, I., 2016. *How should central banks respond to non-neutral inflation expectations?* Bath: University of Bath. *OPUS* [Online]. Available from: <http://opus.bath.ac.uk> [Accessed 4 May 2016].

```
@report{shah.corrick2016hsc,  
  author = {Shah, I. and Corrick, I.},  
  year = {2016},  
  title = {How should central banks respond to non-neutral inflation expectations?},  
  address = {Bath},  
  institution = {University of Bath},  
  library = {OPUS},  
  url = {http://opus.bath.ac.uk},  
  urlyear = {4 May 2016}}
```

RX

- 💡 The Harvard (Bath) style treats preprints as grey literature, and so the **techreport** entry type (or its alias **report**) is the best match semantically.
- 💡 Use **library** to specify the name of the preprint repository. Note that the workaround in **bath.bst** of using **note** instead does not work in **bathx.bst**.

- 📖 Ganju, V., 2021. *A study of EnGeneIC Dream Vectors (EDV's) packaged with the chemotherapy, E-EDV-D682 given simultaneously as non-targeted EDVs carrying an immune enhancer called EDV-GC, in participants with advanced pancreatic and other cancers whose disease has progressed after one or two treatment regimes, or where other standard therapies are not appropriate.* Sydney: University of Sydney. *Australian and New Zealand Clinical Trials Registry* [Online]. Available from: <http://www.anzctr.org.au/Trial/Registration/TrialReview.aspx?id=365258> [Accessed 29 April 2021].
- ⚙ Ganju, V., 2021. *A study of EnGeneIC Dream Vectors (EDV's) packaged with the chemotherapy, E-EDV-D682 given simultaneously as non-targeted EDVs carrying an immune enhancer called EDV-GC, in participants with advanced pancreatic and other cancers whose disease has progressed after one or two treatment regimes, or where other standard therapies are not appropriate.* Sydney: University of Sydney. *Australian and New Zealand Clinical Trials Registry* [Online]. Available from: <http://www.anzctr.org.au/Trial/Registration/TrialReview.aspx?id=365258> [Accessed 29 April 2021].

```
@report{ganju2021sed,
  author = {Ganju, V.},
  year = {2021},
  title = {A study of {EnGeneIC} {Dream} {Vectors} {(EDV's)} packaged with the chemotherapy,
    {E-EDV-D682} given simultaneously as non-targeted {EDVs} carrying an immune enhancer
    called {EDV-GC}, in participants with advanced pancreatic and other cancers whose disease
    has progressed after one or two treatment regimes, or where other standard therapies are
    not appropriate},
  address = {Sydney},
  institution = {University of Sydney},
  library = {Australian and New Zealand Clinical Trials Registry},
  url = {http://www.anzctr.org.au/Trial/Registration/TrialReview.aspx?id=365258},
  urlyear = {29 April 2021}}
```

RX

## Newspaper article

- 🔧 Give the issue's day and month as the month or volume.

- 📖 Haurant, S., 2004. Britain's borrowing hits £1 trillion. *The Guardian*, 29 July, p.16c.
- ⚙ Haurant, S., 2004. Britain's borrowing hits £1 trillion. *The Guardian*, 29 July, p.16c.

```
@article{haurant2004bbh,
  author = {Haurant, S.},
  year = {2004},
  title = {Britain's Borrowing Hits \pounds 1 Trillion},
  journal = {The {Guardian}},
  month = {{29 July}},
  pages = {16c}}
```

RX

- ❗ The extra braces around the month field aren't necessary with Bib<sub>T</sub>EX, but are needed if you want compatibility with biblatex.

- 📖 The Independent, 1992. Picking up the bills. *The Independent*, 4 June, p.28a.
- ⚙ The Independent, 1992. Picking up the bills. *The Independent*, 4 June, p.28a.

```
@article{independent1992pub,
  author = {{The Independent}},
  year = {1992},
  title = {Picking Up the Bills},
  journal = {The {Independent}},
  month = {{4 June}},
  pages = {28a}}
```

RX

📄 Cogley, M., 2020. Corporate confidence slumps to record low. *The Telegraph* [Online], 4 May. Available from: <https://www.telegraph.co.uk/business/2020/05/03/corporate-confidence-slumps-all-time-low/> [Accessed 5 May 2020].

⚙️ Cogley, M., 2020. Corporate confidence slumps to record low. *The Telegraph* [Online], 4 May. Available from: <https://www.telegraph.co.uk/business/2020/05/03/corporate-confidence-slumps-all-time-low/> [Accessed 5 May 2020].

```
@article{cogley2020ccs,  
  author = {Cogley, M.},  
  year = {2020},  
  title = {Corporate Confidence Slumps to Record Low},  
  journal = {{The Telegraph}},  
  volume = {4 May},  
  url = {https://www.telegraph.co.uk/business/2020/05/03/corporate-confidence-slumps-all-time-low/},  
  urlyear = {5 May 2020}}
```

RX

### 5.3 Conference papers

- ❗ As in the standard BibT<sub>E</sub>X styles, `conference` is a legacy (and highly deprecated) alias for `inproceedings`.

#### Conference paper (when proceedings have a named editor)

📄 Crawford, G.I., 1965. Oxygen in metals. In: J.M.A. Lenihan and S.J. Thompson, eds. *Activation analysis: proceedings of a NATO Advanced Study Institute*, 2–4 August 1964, Glasgow. London: Academic Press, pp.113–118.

⚙️ Crawford, G.I., 1965. Oxygen in metals. In: J.M.A. Lenihan and S.J. Thompson, eds. *Activation analysis: Proceedings of a NATO Advanced Study Institute*, 2–4 August 1964, Glasgow. London: Academic Press, pp.113–118.

```
@inproceedings{crawford1965oim,  
  author = {Crawford, G. I.},  
  year = {1965},  
  title = {Oxygen in Metals},  
  editor = {J. M. A. Lenihan and S. J. Thompson},  
  booktitle = {Activation Analysis: Proceedings of a {NATO} {Advanced} {Study} {Institute}},  
  eventyear = {2--4 August 1964},  
  venue = {Glasgow},  
  address = {London},  
  publisher = {Academic Press},  
  pages = {113-118}}
```

RX

Conference paper (when proceedings have no named editor or are part of a major series)

- 📖 Soper, D., 1972. Review of bracken control experiments with asulam. *Proceedings of the 11th British Weed Control Conference*, 15–17 November 1972, Brighton. Brighton: University of Sussex, pp.24–31.
- ⚙️ Soper, D., 1972. Review of bracken control experiments with asulam. *Proceedings of the 11th British Weed Control Conference*, 15–17 November 1972, Brighton. Brighton: University of Sussex, pp.24–31.

```
@inproceedings{soper1972rbc,  
  author = {Soper, D.},  
  year = {1972},  
  title = {Review of Bracken Control Experiments with Asulam},  
  booktitle = {Proceedings of the 11th {British} {Weed} {Control} {Conference}},  
  eventyear = {15--17 November 1972},  
  venue = {Brighton},  
  address = {Brighton},  
  publisher = {University of Sussex},  
  pages = {24-31}}
```

RX

## 5.4 Grey literature

Thesis/dissertation

- 📖 Burrell, J.G., 1973. *The importance of school tours in education*. Thesis (M.A.). Queen's University, Belfast.
- ⚙️ Burrell, J.G., 1973. *The importance of school tours in education*. Thesis (M.A.). Queen's University, Belfast.

```
@mastersthesis{burrell1973ist,  
  author = {Burrell, J. G.},  
  year = {1973},  
  title = {The Importance of School Tours in Education},  
  type = {Thesis (M.A.)},  
  school = {Queen's University, Belfast}}
```

RX



- ❗ Internally, `thesis` is an alias for `phdthesis`. If you need compatibility with `biblatex-bath` and don't want to display a thesis type, use `thesis`.

Report, working paper, or research briefing

- 📖 UNESCO, 1993. *General information programme and UNISIST*. (PGI-93/WS/22). Paris: UNESCO.
- ⚙️ UNESCO, 1993. *General information programme and UNISIST*. (PGI-93/WS/22). Paris: UNESCO.



```
@report{unesco1993gip,  
  author = {{UNESCO}},  
  year = {1993},  
  title = {General Information Programme and {UNISIST}},  
  address = {Paris},  
  institution = {UNESCO},  
  number = {PGI-93/WS/22}}
```

RX

-  BRE, 2007. *Designing quality buildings: a BRE guide*. (Report 497). Bracknell: BRE.
-  BRE, 2007. *Designing quality buildings: a BRE guide*. (Report 497). Bracknell: BRE.



```
@techreport{bre2007dqb,
  author = {{BRE}},
  year = {2007},
  title = {Designing Quality Buildings: a {BRE} Guide},
  address = {Bracknell},
  institution = {BRE},
  type = {Report},
  number = {497}}
```

RX



-  Deneulin, S. and Dinerstein, A.C., 2010. *Hope movements: social movements in the pursuit of human development*. (Bath papers in international development and wellbeing, no. 8). Bath: University of Bath.
-  Deneulin, S. and Dinerstein, A.C., 2010. *Hope movements: Social movements in the pursuit of human development*. (Bath papers in international development and wellbeing, no. 8). Bath: University of Bath.

```
@report{deneulin.dinerstein2010hms,
  author = {Deneulin, S. and Dinerstein, A. C.},
  year = {2010},
  title = {Hope Movements: Social Movements in the Pursuit of Human Development},
  address = {Bath},
  institution = {University of Bath},
  series = {Bath Papers in International Development and Wellbeing},
  number = {no.-8}}
```

RX



-  Internally, `report` is an alias for `techreport`. If you need compatibility with `biblatex-bath` and don't want to display a report type, use `report`.
-  The style inserts a comma between the `series` and `number` fields, but does not insert one between `type` and `number`.

## Standard

-  BSI, 1990. *BS 5605:1990 Recommendations for citing and referencing published material*. London: BSI.
-  BSI, 1990. *BS 5605:1990 Recommendations for citing and referencing published material*. London: BSI.


```
@standard{bs5605:1990,
  author = {{BSI}},
  year = {1990},
  title = {{BS}-5605:1990 {Recommendations} for Citing and Referencing Published Material},
  address = {London},
  organization = {BSI}}
```

RX

-  ASTM, 2019. *ASTM D1655 - 19 Standard specification for aviation turbine fuels*. West Conshohocken, Pa.: ASTM.
-  ASTM, 2019. *ASTM D1655 - 19 Standard specification for aviation turbine fuels*. West Conshohocken, Pa.: ASTM.

```
@standard{astm.d1655,
  author = {{ASTM}},
  year = {2019},
  title = {{ASTM-D1655} - 19 {Standard} Specification for Aviation Turbine Fuels},
  address = {West Conshohocken, Pa.},
  organization = {ASTM}}
```

RX

-  Internally, `standard` is an alias for `manual`.

## Patent

📖 Phillipp Morris Inc., 1981. *Optical perforating apparatus and system*. European patent application 0021165A1. 1981-01-07.

⚙️ Phillipp Morris Inc., 1981. *Optical perforating apparatus and system*. European patent application 0021165A1. 1981-01-07.

```
@patent{pm1981opa,  
  author = {{Phillipp Morris Inc.}},  
  year = {1981},  
  title = {Optical perforating apparatus and system},  
  type = {European patent application},  
  number = {0021165A1. 1981-01-07}}
```

RX

❗ Internally, patent is an alias for manual.

## Unpublished written material and personal communications

📖 Harris, G., 2013. *Focus group recommendations: internal task group report*. Unpublished.

⚙️ Harris, G., 2013. *Focus group recommendations: internal task group report*. Unpublished.

```
@unpublished{harris2013fgr,  
  author = {Harris, G.},  
  year = {2013},  
  title = {Focus group recommendations: internal task group report}}
```

RX

📖 Hadley, S., 2015. *Biomechanics: introductory reading, BM289: sport biomechanics*. University of Bath. Unpublished.

⚙️ Hadley, S., 2015. *Biomechanics: introductory reading, BM289: sport biomechanics*. University of Bath. Unpublished.

```
@unpublished{hadley2015bir,  
  author = {Hadley, S.},  
  year = {2015},  
  title = {Biomechanics: introductory reading, {BM289}: sport biomechanics},  
  howpublished = {University of Bath}}
```

RX

📖 Thomas, D., 2015. Word count and referencing style. *Frequently asked questions discussion board: PHYS 2011: housing studies*. University of Bath. Unpublished.

⚙️ Thomas, D., 2015. Word count and referencing style. *Frequently asked questions discussion board: PHYS 2011: housing studies*. University of Bath. Unpublished.


```
@unpublished{thomas2015wcr,  
  author = {Thomas, D.},  
  year = {2015},  
  title = {Word count and referencing style},  
  booktitle = {Frequently asked questions discussion board: {PHYS} 2011: housing studies},  
  howpublished = {University of Bath}}
```

RX

## 5.5 Audiovisual materials


### Image


 NASA, 2015. *NASA astronaut Tim Kopra on Dec. 21 spacewalk* [Online]. Washington: NASA. Available from: <http://www.nasa.gov/image-feature/nasa-astronaut-tim-kopra-on-dec-21-spacewalk> [Accessed 7 January 2015].


 NASA, 2015. *NASA astronaut Tim Kopra on Dec. 21 spacewalk* [Online]. Washington: NASA. Available from: <http://www.nasa.gov/image-feature/nasa-astronaut-tim-kopra-on-dec-21-spacewalk> [Accessed 7 January 2015].

```
@image{nasa2015nat,  
  author = {{NASA}},  
  year = {2015},  
  title = {{NASA} Astronaut {Tim} {Kopra} on {Dec.\@} 21 Spacewalk},  
  address = {Washington},  
  publisher = {NASA},  
  url = {http://www.nasa.gov/image-feature/nasa-astronaut-tim-kopra-on-dec-21-spacewalk},  
  urlyear = {7 January 2015}}
```

RI


 You can use either the **publisher** or the **organization** field to record the source of the image.


 Iliff, D., 2006. *Royal Crescent in Bath, England - July 2006* [Online]. San Francisco: Wikimedia Foundation. Available from: [https://commons.wikimedia.org/wiki/File:Royal\\_Crescent\\_in\\_Bath,\\_England\\_-\\_July\\_2006.jpg](https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg) [Accessed 7 January 2016].

 Iliff, D., 2006. *Royal Crescent in Bath, England - July 2006* [Online]. San Francisco: Wikimedia Foundation. Available from: [https://commons.wikimedia.org/wiki/File:Royal\\_Crescent\\_in\\_Bath,\\_England\\_-\\_July\\_2006.jpg](https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg) [Accessed 7 January 2016].

```
@image{iliff2006rcb,  
  author = {D. Iliff},  
  year = {2006},  
  title = {{Royal} {Crescent} in {Bath,} {England} - {July} 2006},  
  address = {San Francisco},  
  organization = {Wikimedia Foundation},  
  url = {https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg},  
  urlyear = {7 January 2016}}
```


RI


 Bristol Region Building Record, 1965. *Green Park House (since demolished), viewed from southwest* [Photograph]. BRBR, D/877/1. Archives & Research Collections, University of Bath Library.

 Bristol Region Building Record, 1965. *Green Park House (since demolished), viewed from southwest* [Photograph]. BRBR, D/877/1. Archives & Research Collections, University of Bath Library.

```
@image{brbr1965gph,  
  author = {{Bristol Region Building Record}},  
  year = {1965},  
  title = {{Green} {Park} {House} (since demolished), viewed from southwest},  
  titleaddon = {Photograph},  
  number = {BRBR, D/877/1},  
  library = {Archives & Research Collections, University of Bath Library}}
```

RI

 Use the **library** field to record the archive. If the image is on display rather than archived, use **institution** instead for the museum, gallery or building. If you also wish to provide the name of the organisation that published the image, use the **publisher** field if the location you provide relates to the publisher, but the **organization** field if the location relates to the archive.

 Put archival locator information (e.g. box or shelf number) in the **number** field.



- 💡 Put the clarification of the resource type in `entrysubtype`. In most cases, you can use `titleaddon` instead, but `entrysubtype` will correctly suppress ‘Online’ being shown while `titleaddon` will not.
- ❗ The `@image` entry type can be used for any visual artistic work, including sculptures and interactive exhibits.

## Map

- 📖 Andrews, J. and Dury, A., 1773. *Map of Wiltshire*, 1 inch to 2 miles. Devizes: Wiltshire Record Society.
- ⚙️ Andrews, J. and Dury, A., 1773. *Map of Wiltshire*, 1 inch to 2 miles. Devizes: Wiltshire Record Society.

```
@book{andrews.dury1773wilts,
  author = {Andrews, J. and Dury, A.},
  year = {1773},
  title = {Map of {Wiltshire}},
  series = {1 inch to 2 miles},
  address = {Devizes},
  publisher = {Wiltshire Record Society}}
```

RX

- 📖 Ordnance Survey, 2020. *Street view map of University of Bath* [Online], 1:5000, OS VectorMap® Local. Available from: <https://digimap.edina.ac.uk/roam/map/os> [Accessed 30 April 2020].
- ⚙️ Ordnance Survey, 2020. *Street view map of University of Bath* [Online], 1:5000, OS VectorMap® Local. Available from: <https://digimap.edina.ac.uk/roam/map/os> [Accessed 30 April 2020].

```
@book{os2020bath,
  author = {{Ordnance Survey}},
  year = {2020},
  title = {Street view map of {University of Bath}},
  series = {1:5000, {OS VectorMap® Local}},
  url = {https://digimap.edina.ac.uk/roam/map/os},
  urlyear = {30 April 2020}}
```

RX

- 📖 Google, 2020. *Harbourside, Bristol* [Online], Google Maps. Available from: <https://www.google.co.uk/maps/place/Harbourside,+Bristol/> [Accessed 30 April 2020].
- ⚙️ Google, 2020. *Harbourside, Bristol* [Online], Google Maps. Available from: <https://www.google.co.uk/maps/place/Harbourside,+Bristol/> [Accessed 30 April 2020].

```
@book{google2020harbourside,
  author = {{Google}},
  year = {2020},
  title = {{Harbourside, Bristol}},
  series = {{Google Maps}},
  url = {https://www.google.co.uk/maps/place/Harbourside,+Bristol/},
  urlyear = {30 April 2020}}
```

RX

## Film, video or DVD

- 📺 *Macbeth*, 1948. Film. Directed by Orson Welles. USA: Republic Pictures.
- ⚙️ *Macbeth*, 1948. Film. Directed by Orson Welles. USA: Republic Pictures.

```
@video{macbeth1948,  
  year = {1948},  
  title = {Macbeth},  
  type = {Film},  
  note = {Directed by Orson Welles},  
  address = {USA},  
  publisher = {Republic Pictures}}
```

RX

📌 Internally, `movie` and `video` are aliases for `booklet`.

- 📺 *The elephant man* [Online], 1980. Film. Directed by David Lynch. USA: Brookfilms. Available from: BBC iPlayer [Accessed 4 May 2021].
- ⚙️ *The elephant man* [Online], 1980. Film. Directed by David Lynch. USA: Brookfilms. Available from: BBC iPlayer [Accessed 4 May 2021].

```
@video{elephant-man1980,  
  year = {1980},  
  title = {The elephant man},  
  type = {Film},  
  note = {Directed by David Lynch},  
  address = {USA},  
  publisher = {Brookfilms},  
  library = {BBC iPlayer},  
  urlyear = {4 May 2021}}
```

RX

## Online video/audio

- 📺 Moran, C., 2016. *Save our libraries* [Online]. Available from: <https://youtu.be/gKTfCz4JtVE> [Accessed 29 April 2016].
- ⚙️ Moran, C., 2016. *Save our libraries* [Online]. Available from: <https://youtu.be/gKTfCz4JtVE> [Accessed 29 April 2016].

```
@video{moran2016sol,  
  author = {Moran, C.},  
  year = {2016},  
  title = {Save Our Libraries},  
  url = {https://youtu.be/gKTfCz4JtVE},  
  urlyear = {29 April 2016}}
```

RX

- 📺 Chakrabarti, V., 2016. *How architecture and city planning can combat social inequality* [Online]. Available from: <https://www.curbed.com/2016/5/5/11593058/vishaan-chakrabarti-pau-curbed-appeal-podcast> [Accessed 28 March 2019].
- ⚙️ Chakrabarti, V., 2016. *How architecture and city planning can combat social inequality* [Online]. Available from: <https://www.curbed.com/2016/5/5/11593058/vishaan-chakrabarti-pau-curbed-appeal-podcast> [Accessed 28 March 2019].

```
@audio{chakrabarti2016hac,  
  author = {Chakrabarti, V.},  
  year = {2016},  
  title = {How Architecture and City Planning Can Combat Social Inequality},  
  url = {https://www.curbed.com/2016/5/5/11593058/vishaan-chakrabarti-pau-curbed-appeal-podcast},  
  urlyear = {28 March 2019}}
```

RI

## Television or radio broadcast

- 💡 Use **type** for the medium and **note** for the channel, date and time.
- 💡 If the episode has an individual title, use **series** for the series and **number** for the episode number. Otherwise, give both series and episode number as the **title**.

📺 Hurry up and wait, 2021. *Inside no.9, Episode 6:4*. TV. BBC2, 31 May. 21.30 hrs.  
⚙️ Hurry up and wait, 2021. *Inside no.9, Episode 6:4*. TV. BBC2, 31 May. 21.30 hrs.

```
@video{inn2021huw,  
  year = {2021},  
  title = {Hurry up and wait},  
  series = {Inside No.9},  
  number = {{Episode} 6:4},  
  type = {TV},  
  note = {BBC2, 31 May. 21.30 hrs}}
```

RX

📺 Rick Stein's French odyssey: Episode 5, 2006. TV. BBC2, 23 August. 20.30 hrs.  
⚙️ Rick Stein's French odyssey: Episode 5, 2006. TV. BBC2, 23 August. 20.30 hrs.

```
@video{rsfo2006ep5,  
  year = {2006},  
  title = {Rick {Stein's} {French} Odyssey: {Episode} 5},  
  type = {TV},  
  note = {BBC2, 23 August. 20.30 hrs}}
```

RX

📻 The Archers, 2006. Radio. BBC Radio 4, 23 August. 19.02 hrs.  
⚙️ The Archers, 2006. Radio. BBC Radio 4, 23 August. 19.02 hrs.

```
@audio{archers20060823,  
  year = {2006},  
  title = {The {Archers}},  
  type = {Radio},  
  note = {BBC Radio 4, 23 August. 19.02 hrs}}
```

RX

- 📘 Internally, **audio** and **music** are aliases for **booklet**.

📺 The secret, 2020. *Tiger king: murder, mayhem and madness, Episode 3* [Online]. TV. Available from: Netflix [Accessed 4 May 2021].  
⚙️ The secret, 2020. *Tiger king: Murder, mayhem and madness, Episode 3* [Online]. TV. Available from: Netflix [Accessed 4 May 2021].

```
@video{tkmmm2020ts,  
  year = {2020},  
  title = {The Secret},  
  series = {Tiger King: Murder, Mayhem and Madness},  
  number = {{Episode} 3},  
  type = {TV},  
  library = {Netflix},  
  urlyear = {4 May 2021}}
```

RX

## Music score

- 📖 Beethoven, L. van, 1950. *Symphony no.1 in C, Op.21*. Harmondsworth: Penguin.
- ⚙️ Beethoven, L. van, 1950. *Symphony no.1 in C, Op.21*. Harmondsworth: Penguin.

```
@book{beethoven1950symp1,  
  author = {Ludwig van Beethoven},  
  year = {1950},  
  title = {Symphony no.1 in {C,} {Op.21}},  
  address = {Harmondsworth},  
  publisher = {Penguin}}
```

RX

## 5.6 Digital media

### Website or webpage

While it is normal when using BibTeX to use `misc` for websites, with this style you should use `online` (or `electronic` or `www`) instead. Internally these are all aliases for `manual`.

- 📖 World Health Organization, 2018. *The top 10 causes of death* [Online]. Geneva: World Health Organization. Available from: <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death> [Accessed 29 June 2020].
- ⚙️ World Health Organization, 2018. *The top 10 causes of death* [Online]. Geneva: World Health Organization. Available from: <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death> [Accessed 29 June 2020].

```
@online{who2018ttc,  
  author = {{World Health Organization}},  
  year = {2018},  
  title = {The Top 10 Causes of Death},  
  address = {Geneva},  
  organization = {World Health Organization},  
  url = {https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death},  
  urlyear = {29 June 2020}}
```

RX

- 📖 Manco, J., 2013. *Vernacular architecture* [Online]. Available from: <https://www.buildinghistory.org/style/vernacular.shtml> [Accessed 20 February 2020].
- ⚙️ Manco, J., 2013. *Vernacular architecture* [Online]. Available from: <https://www.buildinghistory.org/style/vernacular.shtml> [Accessed 20 February 2020].

```
@online{manco2013va,  
  author = {Manco, J.},  
  year = {2013},  
  title = {Vernacular Architecture},  
  url = {https://www.buildinghistory.org/style/vernacular.shtml},  
  urlyear = {20 February 2020}}
```

RX

This template is also used for retrievable Generative AI content.

📄 ChatGPT, 2023. *Why is citing and referencing your sources important?* [Online]. San Francisco, Calif.: OpenAI. Available from: <https://chat.openai.com/share/782cb099-a0dc-45b1-8da4-0e99713f2d45> [Accessed 4 September 2023].

⚙️ ChatGPT, 2023. *Why is citing and referencing your sources important?* [Online]. San Francisco, Calif.: OpenAI. Available from: <https://chat.openai.com/share/782cb099-a0dc-45b1-8da4-0e99713f2d45> [Accessed 4 September 2023].

```
@online{chatgpt2023wcr,  
  author = {ChatGPT},  
  year = {2023},  
  title = {Why is Citing and Referencing Your Sources Important?},  
  address = {San Francisco, Calif.},  
  organization = {OpenAI},  
  url = {https://chat.openai.com/share/782cb099-a0dc-45b1-8da4-0e99713f2d45},  
  urlyear = {4 September 2023}}
```

RX

## Social media

📄 Gaiman, N., 2021. *Facebook post* [Online], 21 April. Available from: <https://www.facebook.com/neilgaiman/posts/305187897642814> [Accessed 13 May 2021].

⚙️ Gaiman, N., 2021. *Facebook post* [Online], 21 April. Available from: <https://www.facebook.com/neilgaiman/posts/305187897642814> [Accessed 13 May 2021].

```
@online{gaiman2021fbp,  
  author = {Gaiman, N.},  
  year = {2021},  
  title = {Facebook post},  
  month = {{21 April}},  
  url = {https://www.facebook.com/neilgaiman/posts/305187897642814},  
  urlyear = {13 May 2021}}
```

RX

📄 Library at University of Bath [@bathunilibrary], 2021. *Twitter post* [Online], 10 May. Available from: <https://twitter.com/BathUniLibrary/status/1391774402618998795?s=20> [Accessed 12 May 2021].

⚙️ Library at University of Bath [@bathunilibrary], 2021. *Twitter post* [Online], 10 May. Available from: <https://twitter.com/BathUniLibrary/status/1391774402618998795?s=20> [Accessed 12 May 2021].

```
@online{ublib2021tp,  
  author = {{Library at University of Bath}},  
  nameaddon = {@bathunilibrary},  
  year = {2021},  
  title = {Twitter post},  
  month = {{10 May}},  
  url = {https://twitter.com/BathUniLibrary/status/1391774402618998795?s=20},  
  urlyear = {12 May 2021}}
```

RX

📄 University of Bath Library [@unibathlib], 2021. *Instagram post* [Online], 15 March. Available from: [https://www.instagram.com/p/CMb5GHjFv9A/?utm\\_source=ig\\_web\\_copy\\_link](https://www.instagram.com/p/CMb5GHjFv9A/?utm_source=ig_web_copy_link) [Accessed 7 May 2021].

⚙️ University of Bath Library [@unibathlib], 2021. *Instagram post* [Online], 15 March. Available from: [https://www.instagram.com/p/CMb5GHjFv9A/?utm\\_source=ig\\_web\\_copy\\_link](https://www.instagram.com/p/CMb5GHjFv9A/?utm_source=ig_web_copy_link) [Accessed 7 May 2021].

```
@online{ublib2021ip,  
  author = {{University of Bath Library}},  
  nameaddon = {@unibathlib},  
  year = {2021},  
  title = {Instagram post},  
  month = {{15 March}},  
  url = {https://www.instagram.com/p/CMb5GHjFv9A/?utm_source=ig_web_copy_link},  
  urlyear = {7 May 2021}}
```

RX

### Email discussion lists (jiscmail/listserv etc.)

📄 Clark, T., 5 July 2004. A European UK Libraries Plus? *Lis-link* [Online]. Available from: [lis-link@jiscmail.ac.uk](mailto:lis-link@jiscmail.ac.uk) [Accessed 30 July 2004].

⚙️ Clark, T., 5 July 2004. A European UK Libraries Plus? *Lis-link* [Online]. Available from: [lis-link@jiscmail.ac.uk](mailto:lis-link@jiscmail.ac.uk) [Accessed 30 July 2004].

```
@letter{clark2004euk,  
  author = {Clark, T.},  
  year = {5 July 2004},  
  title = {A {European} {UK} {Libraries} {Plus}?},  
  journal = {Lis-link},  
  url = {lis-link@jiscmail.ac.uk},  
  urlyear = {30 July 2004}}
```

RX

💡 Use the `journal` field to specify the mailing list. Internally, a `letter` entry with a `journal` field is treated like an `article` entry, while one without is treated like an `unpublished` entry.

🔑 You will need to put the full date in the `year` field; unfortunately this means you have to put in extra work to show only the year in citations:

```
\citetext{\citeauthor[2004]{clark2004euk}}
```

(Clark, 2004)

### Database

📄 Bureau van Dijk, 2008. *BT Group plc company report. FAME* [Online]. London: Bureau van Dijk. Available from: <http://www.portal.euromonitor.com> [Accessed 6 November 2014].

⚙️ Bureau van Dijk, 2008. *BT Group plc company report. FAME* [Online]. London: Bureau van Dijk. Available from: <http://www.portal.euromonitor.com> [Accessed 6 November 2014].

```
@dataset{bvd2008bt,  
  author = {{Bureau van Dijk}},  
  year = {2008},  
  title = {{BT} {Group} PLC Company Report},  
  library = {FAME},  
  address = {London},  
  publisher = {Bureau van Dijk},  
  url = {http://www.portal.euromonitor.com},  
  urlyear = {6 November 2014}}
```

RX

💡 Use the `title` field for the entry title, and the `library` field for the name of the database itself.

## Dataset

- 📖 Wilson, D., 2013. *Real geometry and connectedness via triangular description: CAD example bank* [Online]. Bath: University of Bath. Available from: <https://doi.org/10.15125/BATH-00069> [Accessed 20 April 2016].
- ⚙️ Wilson, D., 2013. *Real geometry and connectedness via triangular description: CAD example bank* [Online]. Bath: University of Bath. Available from: <https://doi.org/10.15125/BATH-00069> [Accessed 20 April 2016].

```
@dataset{wilson2013rgc,  
  author = {Wilson, D.},  
  year = {2013},  
  title = {Real Geometry and Connectedness via Triangular Description: {CAD} Example Bank},  
  address = {Bath},  
  publisher = {University of Bath},  
  doi = {10.15125/BATH-00069},  
  urlyear = {20 April 2016}}
```

RX

- 📌 You can use `dataset` instead of `online` as an alias for `manual`.

## Computer program

- 📖 @screencasto, n.d. *Screencast-O-Matic* (v.2) [computer program]. Available from: <https://screencast-o-matic.com/> [Accessed 16 May 2016].
- ⚙️ @screencasto, n.d. *Screencast-O-Matic* (v.2) [computer program]. Available from: <https://screencast-o-matic.com/> [Accessed 16 May 2016].

```
@software{screencasto,  
  author = {@screencasto},  
  title = {{Screencast-O-Matic}},  
  version = {2},  
  entrysubtype = {computer program},  
  url = {https://screencast-o-matic.com/},  
  urlyear = {16 May 2016}}
```

RX

- 📌 Internally, `software` is an alias for `manual`.
- 💡 Put the clarification of the resource type in `entrysubtype`. In most cases, you can use `titleaddon` instead, but `titleaddon` is shown immediately after the title and may be shown alongside the ‘Online’ label, whereas `entrysubtype` is printed after the `version` and supercedes the ‘Online’ label.

## 5.7 Works in languages other than English

### Work in translation

- 📖 Aristotle, 2007. *Nicomachean ethics* (W.D. Ross, Trans.). South Dakota: NuVisions.
- ⚙️ Aristotle, 2007. *Nicomachean ethics* (W.D. Ross, Trans.). South Dakota: NuVisions.

```
@book{aristotle2007ne,  
  author = {Aristotle},  
  year = {2007},  
  title = {Nicomachean Ethics},  
  translator = {W. D. Ross},  
  address = {South Dakota},  
  publisher = {NuVisions}}
```

RX

## Work in the Roman alphabet

📖 Esquivel, L., 2003. *Como agua para chocolate* [Like water for chocolate]. Barcelona: Debolsillo.

⚙️ Esquivel, L., 2003. *Como agua para chocolate* [Like water for chocolate]. Barcelona: Debolsillo.

```
@book{esquivel2003cap,  
  author = {Esquivel, L.},  
  year = {2003},  
  title = {Como Agua para Chocolate},  
  titleaddon = {Like water for chocolate},  
  address = {Barcelona},  
  publisher = {Debolsillo}}
```

RX

💡 Use the `titleaddon` field to supply the English translation of the title.

📖 Thurfjell, W., 1975. Vart har våran doktor tagit vägen? [Where has our doctor gone?]. *Läkartidningen*, 72, p.789.

⚙️ Thurfjell, W., 1975. Vart har våran doktor tagit vägen? [Where has our doctor gone?]. *Läkartidningen*, 72, p.789.

```
@article{thurfjell1975vhv,  
  author = {Thurfjell, W.},  
  year = {1975},  
  title = {Vart har våran doktor tagit vägen?},  
  titleaddon = {Where has our doctor gone?},  
  journal = {Läkartidningen},  
  volume = {72},  
  pages = {789}}
```

RX

## Work in a non-Roman alphabet

📖 Hua, L. 華林甫, 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu 清代以來三峽地區水旱災害的初步研 [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue* 中國社會科學, 1, pp.168–79.

⚙️ Hua, L. 華林甫, 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu 清代以來三峽地區水旱災害的初步研 [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue* 中國社會科學, 1, pp.168–79.

```
@article{hua1999qys1,  
  author = {Linfu \noop{h}華林甫 Hua},  
  year = {1999},  
  title = {Qingdai yilai {Sanxia} diqu shuihan zaihai de chubu yanjiu  
    {清代以來三峽地區水旱災害的初步研}},  
  titleaddon = {A preliminary study of floods and droughts in the {Three} {Gorges} region since  
    the {Qing} dynasty},  
  journal = {Zhongguo shehui kexue \emph{中國社會科學}},  
  volume = {1},  
  pages = {168-79}}
```

RX

💡 The above example tricks BibTeX into treating the original rendering of the author's name as the 'von' part of a Roman-alphabet name. This requires the use of a command that simply gobbles its argument, which you have to define yourself:

```
\newcommand*\noop{[1]{} }
```



For the trick to work, the argument you give to `\noop` must be lowercase, but otherwise it can be anything you like. The trick is not portable to `biblatex-bath`.

- 💡 If the name is due to appear initial first (e.g. after ‘In:’), you can append the non-Roman characters to the author’s surname; to do this, use inverted name order as you would for English double-barrelled names without hyphens, e.g. Hua 華林甫, Linfu.

- 📖 Hua, L., 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue*, 1, pp.168–79.
- ⚙️ Hua, L., 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue*, 1, pp.168–79.

```
@article{hua1999qys2,
  author = {Hua, Linfu},
  year = {1999},
  title = {Qingdai yilai {Sanxia} diqu shuihan zaihai de chubu yanjiu},
  titleaddon = {A preliminary study of floods and droughts in the {Three} {Gorges} region since
    the {Qing} dynasty},
  journal = {Zhongguo shehui kexue},
  volume = {1},
  pages = {168-79}}
```

RX

- 📖 Pamporov, A., 2006. *Romskoto vsekidnevie v Balgariya* [Roma everyday life in Bulgaria]. Veliko Tarnovo: Faber.
- ⚙️ Pamporov, A., 2006. *Romskoto vsekidnevie v Balgariya* [Roma everyday life in Bulgaria]. Veliko Tarnovo: Faber.

```
@book{pamporov2006rvb,
  author = {Pamporov, A.},
  year = {2006},
  title = {Romskoto vsekidnevie v {Balgariya}},
  titleaddon = {Roma everyday life in Bulgaria},
  address = {Veliko Tarnovo},
  publisher = {Faber}}
```

RX

## 5.8 Government and parliamentary document

### House of Commons paper

- 💡 Use this form for reports of House of Commons select committees.

- 📖 Great Britain. Parliament. House of Commons, 2004. *National Savings investment deposits: account 2002–2003*. (HC 2003/04, 30). London: National Audit Office.
- ⚙️ Great Britain. Parliament. House of Commons, 2004. *National Savings investment deposits: account 2002–2003*. (HC 2003/04, 30). London: National Audit Office.

```
@report{gb.hc2003/04-30,
  author = {{Great Britain. Parliament. House of Commons}},
  year = {2004},
  title = {National {Savings} Investment Deposits: account 2002--2003},
  address = {London},
  publisher = {National Audit Office},
  series = {{HC} 2003/04},
  number = {30}}
```

RX

- 💡 To achieve compatibility with `biblatex-bath`, you can either set the `type` field to the empty string or use the `report` entry type, which is an alias for `techreport`.

## House of Lords paper

💡 Use this form for reports of House of Lords select committees.

📖 Great Britain. Parliament. House of Lords, 1987. *Social fund (maternity and funeral expenses) bill*. (HL 1986/87, (66)). London: HMSO.

⚙️ Great Britain. Parliament. House of Lords, 1987. *Social fund (maternity and funeral expenses) bill*. (HL 1986/87, (66)). London: HMSO.

```
@report{gb.hl1986/87-66,
  author = {{Great Britain. Parliament. House of Lords}},
  year = {1987},
  title = {Social Fund (Maternity and Funeral Expenses) Bill},
  address = {London},
  publisher = {HMSO},
  series = {{HL} 1986/87},
  number = {66}}
```

RX

🔑 For joint committees, you will have to hack this slightly, putting the session years in **series** and the HL and HC numbers in **number**.

## Command paper

📖 Great Britain. Ministry of Defence, 2004. *Delivering security in a changing world: defence white paper*. (Cm. 6041). London: TSO.

⚙️ Great Britain. Ministry of Defence, 2004. *Delivering security in a changing world: defence white paper*. (Cm. 6041). London: TSO.

```
@report{gb.cm6041,
  author = {{Great Britain. Ministry of Defence}},
  year = {2004},
  title = {Delivering Security in a Changing World{:} Defence White Paper},
  address = {London},
  publisher = {TSO},
  series = {Cm},
  number = {6041}}
```

RX

💡 To support the Command Paper template, if you give a report the **series** ‘C’, ‘Cd’, ‘Cmd’, ‘Cmnd’ or ‘Cm’, it is followed by a dot rather than a comma. (If you need the comma as normal, wrap the value in an extra pair of braces.)

If you prefer, you can give the Command Paper abbreviation in **type** and supply the dot yourself.

## House of Commons/House of Lords bill



📖 Great Britain. Parliament. House of Commons, 1988. *Local government finance bill*. (Bills | 1987/88, 66). London: HMSO.

⚙️ Great Britain. Parliament. House of Commons, 1988. *Local government finance bill*. (Bills | 1987/88, 66). London: HMSO.

```
@legislation{gb.bill1987/88-66,
  author = {{Great Britain. Parliament. House of Commons}},
  year = {1988},
  title = {Local Government Finance Bill},
  address = {London},
  publisher = {HMSO},
  series = {{Bills | 1987/88}},
  number = {66}}
```

RX



## Act of Parliament (UK Statutes) before 1963

-  *Witchcraft Act 1735* (9 Geo.2, c.5).
-  *Witchcraft Act 1735* (9 Geo.2, c.5).

```
@legislation{gb.wa1735,
  title = {Witchcraft {Act}},
  year = {1735},
  series = {9 Geo.2},
  chapter = {5}}
```

RX



## Act of Parliament (UK Statutes) 1963 onwards

-  *Pensions Act 2014*, c.19. London: TSO.
-  *Pensions Act 2014*, c.19. London: TSO.

```
@legislation{gb.pa2014,
  title = {Pensions {Act}},
  year = {2014},
  chapter = {19},
  address = {London},
  publisher = {TSO}}
```

RX

## Statutory instrument

-  *The Human Medicines Regulations 2012* [Online], No.1916, United Kingdom: HMSO. Available from: [http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi\\_20121916\\_en.pdf](http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi_20121916_en.pdf) [Accessed 17 April 2016].
-  *The Human Medicines Regulations 2012* [Online], No.1916, United Kingdom: HMSO. Available from: [http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi\\_20121916\\_en.pdf](http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi_20121916_en.pdf) [Accessed 17 April 2016].

```
@legislation{gb.hmr2012,
  entrysubtype = {secondary},
  title = {The {Human} {Medicines} {Regulations}},
  year = {2012},
  number = {No.1916},
  address = {United Kingdom},
  publisher = {HMSO},
  url = {http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi_20121916_en.pdf},
  urlyear = {17 April 2016}}
```

RX

💡 Use the entrysubtype 'secondary' to put the number in the right place.

## Parliamentary debate

- 📖 Great Britain. Parliament. House of Commons, 2024. Rwanda Plan Cost and Asylum System. *Hansard* [Online], 744, 9 January 2024. Available from: <https://hansard.parliament.uk/Commons/2024-01-09/debates/57105D77-72DE-4246-9F6E-ED87FCB9A088/RwandaPlanCostAndAsylumSystem> [Accessed 12 December 2024].
- ⚙️ Great Britain. Parliament. House of Commons, 2024. Rwanda Plan Cost and Asylum System. *Hansard* [Online], 744, 9 January 2024. Available from: <https://hansard.parliament.uk/Commons/2024-01-09/debates/57105D77-72DE-4246-9F6E-ED87FCB9A088/RwandaPlanCostAndAsylumSystem> [Accessed 12 December 2024].

```
@legal{gb.hc2024rpc,  
  entrysubtype = {parliamentary},  
  author = {{Great Britain. Parliament. House of Commons}},  
  year = {2024},  
  title = {{Rwanda Plan Cost and Asylum System}},  
  journal = {Hansard},  
  volume = {744},  
  eventyear = {9 January 2024},  
  url = {https://hansard.parliament.uk/Commons/2024-01-09/debates/57105D77-72DE-4246-9F6E-ED87FCB9A088/RwandaPlanCostAndAsylumSystem},  
  urlyear = {12 December 2024}}
```

RX

## 5.9 Legal document: EU reports/legislation

### EU publication

The following example has been removed.

- 📖 European Commission, 2015. *General report on the activities of the European Union 2014*. Luxembourg: Publications Office of the European Union.
- ⚙️ European Commission, 2015. *General report on the activities of the European Union 2014*. Luxembourg: Publications Office of the European Union.

```
@report{ec2015gra,  
  author = {{European Commission}},  
  year = {2015},  
  title = {General Report on the Activities of the {European} {Union} 2014},  
  address = {Luxembourg},  
  publisher = {Publications Office of the European Union}}
```

RX

### EU regulation or directive, decision, recommendation or opinion

- 📖 Council Regulation (EC) 1984/2003 of 8 April 2003 introducing a system for the statistical monitoring of trade in bluefin tuna, swordfish and big eye tuna within the Community [2003] *OJ* L295.
- ⚙️ Council Regulation (EC) 1984/2003 of 8 April 2003 introducing a system for the statistical monitoring of trade in bluefin tuna, swordfish and big eye tuna within the Community [2003] *OJ* L295.

```
@legislation{eu.dir1984/2003,  
  title = {Council {Regulation} ({EC}) 1984/2003 of 8 {April} 2003 Introducing a System for the Statistical Monitoring of Trade in Bluefin Tuna, Swordfish and Big Eye Tuna within the {Community}},  
  year = {2003},  
  journal = {OJ},  
  series = {L},  
  volume = {295}}
```

RX

🔑 Use `\defcitealias` to provide a suitable citation string:

```
\defcitealias{eu.dir1984/2003}{%
  Council Regulation [EC] 1984/2003}
\citepalias{eu.dir1984/2003}
```

(Council Regulation [EC] 1984/2003)


## 5.10 Legal document: case reports

### Judgment of the European Court of Justice

-  *Alessandrini Srl and others v. Commission* (C-295/03 P) [2005] ECR I-5700.
-  *Alessandrini Srl and others v. Commission* (C-295/03 P) [2005] ECR I-5700.

```
@jurisdiction{srl.etal-v-comm2005,
  title = {Alessandrini {Srl} and others v.~{Commission}},
  number = {C-295/03 P},
  year = {2005},
  journal = {ECR},
  volume = {I},
  pages = {5700}}
```

RX



-  Use the `number` field (or the non-standard `casenumber` field) for the case number. For Commission Decisions, use the (non-standard) `casenumber` or (`blatex-oscola`) `userb` field for the Commission case number, `number` for the formal decision number, and give ‘Commission’ as the institution.

### Legal case study

-  *Seldon v Clarkson Wright & Jakes*, 2009. England and Wales Court of Appeal (Civil Division), 889. *Bailii* [Online]. Available from: <http://www.bailii.org/ew/cases/EWCA/Civ/2009/889.html> [Accessed 23 April 2024].
-  *Seldon v Clarkson Wright & Jakes*, 2009. England and Wales Court of Appeal (Civil Division), 889. *Bailii* [Online]. Available from: <http://www.bailii.org/ew/cases/EWCA/Civ/2009/889.html> [Accessed 23 April 2024].

```
@jurisdiction{seldon-v-c.w.j2009,
  title = {Seldon v-~{Clarkson} {Wright} \& {Jakes}},
  year = {2009},
  institution = {England and Wales Court of Appeal (Civil Division)},
  number = {889},
  journal = {Bailii},
  url = {http://www.bailii.org/ew/cases/EWCA/Civ/2009/889.html},
  urlyear = {23 April 2024}}
```

RX

-  *Braceurself Ltd v NHS England*, 2013. England and Wales Court of Appeal (Civil Division), 39. *Construction law reports*, 212, pp.95–135.
-  *Braceurself Ltd v NHS England*, 2013. England and Wales Court of Appeal (Civil Division), 39. *Construction law reports*, 212, pp.95–135.

```
@jurisdiction{braceurself-v-nhs.england2013,
  title = {{Braceurself} {Ltd} v-~{NHS} {England}},
  year = {2013},
  institution = {England and Wales Court of Appeal (Civil Division)},
  number = {39},
  journal = {Construction law reports},
  volume = {212},
  pages = {95-135}}
```

RX

It is also possible to achieve a more traditional or **neutral citation** format, similar to Harvard (Bath)’s pre-2024 format:

📖 *Seldon v Clarkson Wright & Jakes* [2012] UKSC 16.  
⚙️ *Seldon v Clarkson Wright & Jakes* [2012] UKSC 16.

```
@jurisdiction{seldon-v-c.w.j2012,  
  title = {Seldon v~{Clarkson} {Wright} \& {Jakes}},  
  year = {2012},  
  journal = {UKSC},  
  pages = {16}}
```

RX

❗ Generally speaking, in a neutral citation the year should be in square brackets if it is essential to the citation (unless it is a Scottish case, in which case it is printed bare), and in parentheses if it is not.

💡 By default, the style assumes the year is essential if and only if a volume number is *not* provided; to override this, you can use the `year-essential` option:

```
options = {year-essential=true},
```

To use Scottish style for a case, you can either use the `scottish-style` option or the keyword `sc`:

```
options = {scottish-style},  
keywords = {sc},
```

❗ Please bear in mind that the `options` and `keywords` fields from `biblatex` have been emulated rather crudely. You can only supply one of each and the text must match exactly (so don't add spaces or anything like that).

🔑 This should cover most cases, but legal references tend to enforce their own conventions no matter what the rest of the reference list is doing, and it is out of scope for this style to cater for every variation. Therefore if you need a different format (e.g. for an American case), you may prefer to format the reference more-or-less by hand:

```
title = {Seldon v~{Clarkson} {Wright} \& {Jakes}},  
sortyear = {2009},  
note = {[2009] EWCA Civ 889}
```

## References

- Adams, G., 2009. *Test citation one*. London: Imperial College Bookstall.
- Adams, G., 2014. *Test citation two*. Oxford: Oxford University Press.
- Adams, G., 2017. *Test citation three*. Cambridge: Cambridge University Press.
- Alessandrini Srl and others v. Commission* (C-295/03 P) [2005] ECR I-5700.
- Andrews, J. and Dury, A., 1773. *Map of Wiltshire*, 1 inch to 2 miles. Devizes: Wiltshire Record Society.
- The Archers, 2006. Radio. BBC Radio 4, 23 August. 19.02 hrs.
- Aristotle, 2007. *Nicomachean ethics* (W.D. Ross, Trans.). South Dakota: NuVisions.
- ASTM, 2019. *ASTM D1655 - 19 Standard specification for aviation turbine fuels*. West Conshohocken, Pa.: ASTM.
- Beethoven, L. van, 1950. *Symphony no.1 in C, Op.21*. Harmondsworth: Penguin.

- Blockley, D., 2021. *Engineering: a very short introduction* [Online]. Oxford: Oxford University Press. Available from: <https://www.amazon.co.uk/kindle> [Accessed 30 June 2021].
- Braceurself Ltd v NHS England*, 2013. England and Wales Court of Appeal (Civil Division), 39. *Construction law reports*, 212, pp.95–135.
- BRE, 2007. *Designing quality buildings: a BRE guide*. (Report 497). Bracknell: BRE.
- Bristol Region Building Record, 1965. *Green Park House (since demolished), viewed from southwest* [Photograph]. BRBR, D/877/1. Archives & Research Collections, University of Bath Library.
- British National Formulary, 2019. *Aspirin* [Online]. London: Pharmaceutical Press. Available from: [https://www.medicinescomplete.com/#/content/bnf/\\_456850132](https://www.medicinescomplete.com/#/content/bnf/_456850132) [Accessed 26 November 2019].
- British National Formulary, 2020a. 79th ed. *Aspirin*. London: Pharmaceutical Press, pp.280–281.
- British National Formulary, 2020b. 79th ed. London: Pharmaceutical Press.
- BSI, 1990. *BS 5605:1990 Recommendations for citing and referencing published material*. London: BSI.
- Burchard, J.E., 1965. How humanists use a library. In: C.F.J. Overhage and J.R. Harman, eds. *Intrex: report on a planning conference and information transfer experiments*. Cambridge, Mass.: MIT Press, pp.41–87.
- Bureau van Dijk, 2008. *BT Group plc company report. FAME* [Online]. London: Bureau van Dijk. Available from: <http://www.portal.euromonitor.com> [Accessed 6 November 2014].
- Burrell, J.G., 1973. *The importance of school tours in education*. Thesis (M.A.). Queen’s University, Belfast.
- Chakrabarti, V., 2016. *How architecture and city planning can combat social inequality* [Online]. Available from: <https://www.curbed.com/2016/5/5/11593058/vishaan-chakrabarti-pau-curbed-appeal-podcast> [Accessed 28 March 2019].
- ChatGPT, 2023. *Why is citing and referencing your sources important?* [Online]. San Francisco, Calif.: OpenAI. Available from: <https://chat.openai.com/share/782cb099-a0dc-45b1-8da4-0e99713f2d45> [Accessed 4 September 2023].
- Clark, T., 5 July 2004. A European UK Libraries Plus? *Lis-link* [Online]. Available from: [lis-link@jiscmail.ac.uk](mailto:lis-link@jiscmail.ac.uk) [Accessed 30 July 2004].
- Cogley, M., 2020. Corporate confidence slumps to record low. *The Telegraph* [Online], 4 May. Available from: <https://www.telegraph.co.uk/business/2020/05/03/corporate-confidence-slumps-all-time-low/> [Accessed 5 May 2020].
- Council Regulation (EC) 1984/2003 of 8 April 2003 introducing a system for the statistical monitoring of trade in bluefin tuna, swordfish and big eye tuna within the Community [2003] *OJ* L295.
- Crawford, G.I., 1965. Oxygen in metals. In: J.M.A. Lenihan and S.J. Thompson, eds. *Activation analysis: Proceedings of a NATO Advanced Study Institute, 2–4 August 1964*, Glasgow. London: Academic Press, pp.113–118.
- Deneulin, S. and Dinerstein, A.C., 2010. *Hope movements: Social movements in the pursuit of human development*. (Bath papers in international development and wellbeing, no. 8). Bath: University of Bath.
- Devlin, S.M., Martin, A. and Ostrovnya, I., 2021. Identifying prognostic pairwise relationships among bacterial species in microbiome studies. *PLOS computational biology* [Online], 17(11). Available from: <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501> [Accessed 9 December 2021].

*The elephant man* [Online], 1980. Film. Directed by David Lynch. USA: Brookfilms. Available from: BBC iPlayer [Accessed 4 May 2021].

Esquivel, L., 2003. *Como agua para chocolate* [Like water for chocolate]. Barcelona: Debolsillo.

European Commission, 2015. *General report on the activities of the European Union 2014*. Luxembourg: Publications Office of the European Union.

Gaiman, N., 2021. *Facebook post* [Online], 21 April. Available from: <https://www.facebook.com/neilgaiman/posts/305187897642814> [Accessed 13 May 2021].

Ganju, V., 2021. *A study of EnGeneIC Dream Vectors (EDV's) packaged with the chemotherapy, E-EDV-D682 given simultaneously as non-targeted EDVs carrying an immune enhancer called EDV-GC, in participants with advanced pancreatic and other cancers whose disease has progressed after one or two treatment regimes, or where other standard therapies are not appropriate*. Sydney: University of Sydney. *Australian and New Zealand Clinical Trials Registry* [Online]. Available from: <http://www.anzctr.org.au/Trial/Registration/TrialReview.aspx?id=365258> [Accessed 29 April 2021].

Google, 2020. *Harbourside, Bristol* [Online], Google Maps. Available from: <https://www.google.co.uk/maps/place/Harbourside,+Bristol/> [Accessed 30 April 2020].

Great Britain. Ministry of Defence, 2004. *Delivering security in a changing world: defence white paper*. (Cm. 6041). London: TSO.

Great Britain. Parliament. House of Commons, 1988. *Local government finance bill*. (Bills | 1987/88, 66). London: HMSO.

Great Britain. Parliament. House of Commons, 2004. *National Savings investment deposits: account 2002–2003*. (HC 2003/04, 30). London: National Audit Office.

Great Britain. Parliament. House of Commons, 2024. *Rwanda Plan Cost and Asylum System*. *Hansard* [Online], 744, 9 January 2024. Available from: <https://hansard.parliament.uk/Commons/2024-01-09/debates/57105D77-72DE-4246-9F6E-ED87FCB9A088/RwandaPlanCostAndAsylumSystem> [Accessed 12 December 2024].

Great Britain. Parliament. House of Lords, 1987. *Social fund (maternity and funeral expenses) bill*. (HL 1986/87, (66)). London: HMSO.

Hadley, S., 2015. *Biomechanics: introductory reading, BM289: sport biomechanics*. University of Bath. Unpublished.

Harris, G., 2013. *Focus group recommendations: internal task group report*. Unpublished.

Haurant, S., 2004. Britain's borrowing hits £1 trillion. *The Guardian*, 29 July, p.16c.

Haynes, W.M., ed., 2014. *CRC handbook of chemistry and physics* [Online]. 94th ed. Boca Raton, Fla.: CRC Press/Taylor and Francis. Available from: <http://www.hbcnetbase.com> [Accessed 16 June 2016].

Hua, L., 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue*, 1, pp.168–79.

Hua, L. 華林甫, 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu 清代以來三峽地區水旱災害的初步研 [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue* 中國社會科學, 1, pp.168–79.

*The Human Medicines Regulations 2012* [Online], No.1916, United Kingdom: HMSO. Available from: [http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi\\_20121916\\_en.pdf](http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi_20121916_en.pdf) [Accessed 17 April 2016].

Hurry up and wait, 2021. *Inside no.9, Episode 6:4*. TV. BBC2, 31 May. 21.30 hrs.



- Iliff, D., 2006. *Royal Crescent in Bath, England - July 2006* [Online]. San Francisco: Wikimedia Foundation. Available from: [https://commons.wikimedia.org/wiki/File:Royal\\_Crescent\\_in\\_Bath,\\_England\\_-\\_July\\_2006.jpg](https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg) [Accessed 7 January 2016].
- Library at University of Bath [@bathunilibrary], 2021. *Twitter post* [Online], 10 May. Available from: <https://twitter.com/BathUniLibrary/status/1391774402618998795?s=20> [Accessed 12 May 2021].
- Liontou, C., Kontopodis, E., Oikonomidis, N., Maniotis, C., Tassopoulos, A., Tsiafoutis, I., Lazaris, E. and Koutouzis, M., 2019. Distal radial access: a review article. *Cardiovascular revascularization medicine* [Online], in press. Available from: <https://www.sciencedirect.com/science/article/pii/S1553838919303367> [Accessed 19 June 2019].
- Macbeth*, 1948. Film. Directed by Orson Welles. USA: Republic Pictures.
- Manco, J., 2013. *Vernacular architecture* [Online]. Available from: <https://www.buildinghistory.org/style/vernacular.shtml> [Accessed 20 February 2020].
- Moran, C., 2016. *Save our libraries* [Online]. Available from: <https://youtu.be/gKTfCz4JtVE> [Accessed 29 April 2016].
- NASA, 2015. *NASA astronaut Tim Kopra on Dec. 21 spacewalk* [Online]. Washington: NASA. Available from: <http://www.nasa.gov/image-feature/nasa-astronaut-tim-kopra-on-dec-21-spacewalk> [Accessed 7 January 2015].
- Newman, R., 2010. Malaria control beyond 2010. *British medical journal*, 341(7765), pp.157–208.
- Open University, 1972. *Electricity and magnetism*. Bletchley: Open University Press.
- Ordnance Survey, 2020. *Street view map of University of Bath* [Online], 1:5000, OS VectorMap@Local. Available from: <https://digimap.edina.ac.uk/roam/map/os> [Accessed 30 April 2020].
- Pamporov, A., 2006. *Romskoto vsekidnevie v Balgariya* [Roma everyday life in Bulgaria]. Veliko Tarnovo: Faber.
- Pensions Act 2014*, c.19. London: TSO.
- Phillipp Morris Inc., 1981. *Optical perforating apparatus and system*. European patent application 0021165A1. 1981-01-07.
- Rang, H.P., Dale, M.M., Ritter, J.M., Flower, R.J. and Henderson, G., 2012. *Rang and Dale's pharmacology*. 7th ed. Edinburgh: Elsevier Churchill Livingstone.
- Reid, D.R., 1967. Physical testing of polymer films. In: S.H. Pinner, ed. *Modern packaging films*. London: Butterworths, pp.143–183.
- Rick Stein's French odyssey: Episode 5, 2006. TV. BBC2, 23 August. 20.30 hrs.
- Rothman, K.J., Greenland, S. and Lash, T.L., eds, 2008. *Modern epidemiology*. 3rd ed. Philadelphia, Pa.: Lippincott Williams & Wilkins.
- @screencasto, n.d. *Screencast-O-Matic* (v.2) [computer program]. Available from: <https://screencast-o-matic.com/> [Accessed 16 May 2016].
- The secret, 2020. *Tiger king: Murder, mayhem and madness, Episode 3* [Online]. TV. Available from: Netflix [Accessed 4 May 2021].
- Seldon v Clarkson Wright & Jakes*, 2009. England and Wales Court of Appeal (Civil Division), 889. *Bailii* [Online]. Available from: <http://www.bailii.org/ew/cases/EWCA/Civ/2009/889.html> [Accessed 23 April 2024].
- Seldon v Clarkson Wright & Jakes* [2012] UKSC 16.
- Shah, I. and Corrick, I., 2016. *How should central banks respond to non-neutral inflation expectations?* Bath: University of Bath. *OPUS* [Online]. Available from: <http://opus.bath.ac.uk> [Accessed 4 May 2016].

- Solomon, M.R., Askegaard, S., Hogg, M. and Bamossy, G.J., 2019. *Consumer behaviour: a European perspective*. 7th ed. Harlow: Pearson.
- Soper, D., 1972. Review of bracken control experiments with asulam. *Proceedings of the 11th British Weed Control Conference*, 15–17 November 1972, Brighton. Brighton: University of Sussex, pp.24–31.
- Steward, S., Connelly, D. and Robinson, J., 2020. Everything you should know about the coronavirus outbreak. *The pharmaceutical journal* [Online]. Available from: <https://www.pharmaceutical-journal.com/news-and-analysis/features/everything-you-should-know-about-the-coronavirus-outbreak/20207629.article> [Accessed 30 April 2020].
- Stieg, M.F., 1981a. Continuing education and the reference librarian in the academic and research library. *Library journal*, 105(22), pp.2547–2551.
- Stieg, M.F., 1981b. The information needs of historians. *College and research libraries*, 42(6), pp.549–560.
- The Independent, 1992. Picking up the bills. *The Independent*, 4 June, p.28a.
- Thomas, D., 2015. Word count and referencing style. *Frequently asked questions discussion board: PHYS 2011: housing studies*. University of Bath. Unpublished.
- Thurfjell, W., 1975. Vart har våran doktor tagit vägen? [Where has our doctor gone?]. *Läkartidningen*, 72, p.789.
- UNESCO, 1993. *General information programme and UNISIST*. (PGI-93/WS/22). Paris: UNESCO.
- University of Bath Library [@unibathlib], 2021. *Instagram post* [Online], 15 March. Available from: [https://www.instagram.com/p/CMb5GHjFv9A/?utm\\_source=ig\\_web\\_copy\\_link](https://www.instagram.com/p/CMb5GHjFv9A/?utm_source=ig_web_copy_link) [Accessed 7 May 2021].
- Wetzstein, G., Ozcan, A., Gigan, S., Fan, S., Englund, D., Soljagic, M., Denz, C., Miller, D.A.B. and Psaltis, D., 2020. Inference in artificial intelligence with deep optics and photonics. *Nature*, 588(7836), pp.39–47.
- Wilson, D., 2013. *Real geometry and connectedness via triangular description: CAD example bank* [Online]. Bath: University of Bath. Available from: <https://doi.org/10.15125/BATH-00069> [Accessed 20 April 2016].
- Witchcraft Act 1735* (9 Geo.2, c.5).
- World Health Organization, 2018. *The top 10 causes of death* [Online]. Geneva: World Health Organization. Available from: <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death> [Accessed 29 June 2020].

## 6 Licence

Copyright 2016-2025 University of Bath.

This work consists of the documented LaTeX file bath-bst.dtx and a Makefile.

The text files contained in this work may be distributed and/or modified under the conditions of the [LaTeX Project Public License \(LPPL\)](#), either version 1.3c of this license or (at your option) any later version.

This work is ‘maintained’ (as per LPPL maintenance status) by [Alex Ball](#).

## 7 Implementation

The line numbers for the listings below match `bathx.bst`.

### 7.1 Data model

We begin by listing the entry fields that we will recognize. These are the standard Bib<sub>T</sub>E<sub>X</sub> ones plus `archive`, `doi`, `eid`, `eprint`, `entrysubtype`, `eventyear`, `keywords`, `language`, `library`, `options`, `pubstate`, `sortyear`, `titleaddon`, `translator`, `url`, `urldate`, `urlyear`, `userb`, `venue`, and `version`, all of which have been backported from `natbib` and `biblatex`. `casenumber` is specific to this style.

```
16 ENTRY
17 { address
18   archive
19   author
20   booktitle
21   casenumber
22   chapter
23   doi
24   edition
25   editor
26   eid
27   entrysubtype
28   eprint
29   eventyear
30   howpublished
31   institution
32   journal
33   key
34   keywords
35   language
36   library
37   month
38   nameaddon
39   note
40   number
41   options
42   organization
43   pages
44   publisher
45   pubstate
46   school
47   series
48   sortyear
49   title
50   titleaddon
51   translator
52   type
53   url
54   urldate
55   urlyear
56   userb
57   venue
58   version
59   volume
60   year
61 }
62 {}
63 { label extra.label sort.label short.list }
```

### 7.2 Writing out references to a file

This is the first of three sections that deal with writing out to the `.bbl` bibliography file. That file contains the `thebibliography` environment, which in turn contains `\bibitem` commands each providing citation data and followed by a formatted reference. The business of writing the environment, sorting the items and calculating the first half of the citation data for each item is covered in section 7.8. Calculating the second half of the citation data and writing the `\bibitem` command is

covered in section 7.5; some of the functions defined there are used in section 7.8, hence the backward structure. This section contains generic functions for writing out the formatted reference.

We define a set of integers for saving an output state. We will use them to implement the mechanism for stringing the elements of the reference together. Whenever an element is printed, it is separated from any previous previous text by a delimiter, and the delimiter chosen depends on the state. We define five states which trigger increasingly strong delimiters.

```

64 INTEGERS {
65   output.state before.all mid.clause mid.sentence after.sentence after.block
66   online.shown
67 }
68 FUNCTION {init.state.consts}
69 { #0 'before.all :=
70   #1 'mid.clause :=
71   #2 'mid.sentence :=
72   #3 'after.sentence :=
73   #4 'after.block :=
74 }

```

We also define two variables to store strings in.

```

75 STRINGS { s t }

```

The `output.nonnull` function takes the top two tokens (strings) from the stack. It appends the appropriate delimiter to the earlier token and writes it to the `.bbl` file. It then returns the later (topmost) token to the top of the stack ready for the next occurrence of the function. These are the delimiters used in each state:

- `before.all` = nothing is written.
- `mid.clause` = a space is written.
- `mid.sentence` = a comma is written.
- `after.sentence` = a period is written.
- `after.block` = a period, line break and `\newblock` are written.

Afterwards, the output state is set to `mid.sentence`.

```

76 FUNCTION {output.nonnull}
77 { 's :=
78   output.state mid.sentence =
79   { ", " * write$ }
80   { output.state after.sentence =
81     { add.period$ " " * write$ }
82     { output.state before.all =
83       'write$
84       { output.state mid.clause =
85         { " " * write$ }
86         { add.period$ write$
87           newline$
88           "\newblock " write$
89         }
90       } if$
91     } } if$
92   } } if$
93 }
94 if$
95 mid.sentence 'output.state :=
96 }
97 if$
98 s
99 }

```

The ‘after’ output states persist if nothing is printed. The ‘mid.clause’ state shouldn’t, so we provide a macro for reverting to ‘mid.sentence’.

```

100 FUNCTION {end.clause}
101 { output.state mid.clause =
102   { mid.sentence 'output.state := }
103   'skip$
104   if$
105 }

```

We provide two wrappers around `output.nonnull` to avoid passing it empty strings: `output` is silent, while `output.check` will write a warning to the log that the field is empty. The latter consumes an additional string (at the top of the stack) for printing in the warning message.

```

106 FUNCTION {output}
107 { duplicate$ empty$
108   { pop$ end.clause }
109   'output.nonnull
110   if$
111 }
112 FUNCTION {output.check}
113 { 't :=
114   duplicate$ empty$
115   { pop$ "empty " t * " in " * cite$ * warning$
116     end.clause
117   }
118   'output.nonnull
119   if$
120 }

```

We provide some convenience functions for changing the state from `mid.sentence`. The order of precedence is `before.all`, `after.block`, `after.sentence`, `mid.clause/mid.sentence`.

```

121 FUNCTION {new.block}
122 { output.state before.all =
123   'skip$
124   { after.block 'output.state := }
125   if$
126 }
127 FUNCTION {new.sentence}
128 { output.state after.block =
129   'skip$
130   { output.state before.all =
131     'skip$
132     { after.sentence 'output.state := }
133     if$
134   }
135   if$
136 }
137 FUNCTION {continue.clause}
138 { output.state after.sentence =
139   'skip$
140   { output.state after.block =
141     'skip$
142     { output.state before.all =
143       'skip$
144       { mid.clause 'output.state := }
145       if$
146     }
147     if$
148   }
149   if$
150 }

```

The `date.block` function is a handy alias.

```

151 FUNCTION {date.block}
152 { new.block
153 }

```

The `bibinfo` functions consume two tokens – the value and name of a field respectively – and return a single string. If the value is non-null, the tokens are concatenated in the form `\bibinfo{field name}{field value}`, otherwise an empty string is returned.

The `check` version does this silently, while the `warn` version issues a warning in the log file if the value is missing or empty.

```

154 FUNCTION {bibinfo.check}
155 { swap$
156   duplicate$ missing$
157   { pop$ pop$
158     ""
159   }
160   { duplicate$ empty$
161     { swap$ pop$
162       }
163     { swap$
164       "\bibinfo{" swap$ * "}" * swap$ * "}" *
165     }
166     if$
167   }
168   if$
169 }
170 FUNCTION {bibinfo.warn}
171 { swap$
172   duplicate$ missing$
173   { swap$ "missing " swap$ * " in " * cite$ * warning$ pop$
174     ""
175   }
176   { duplicate$ empty$
177     { swap$ "empty " swap$ * " in " * cite$ * warning$
178       }
179     { swap$
180       "\bibinfo{" swap$ * "}" * swap$ * "}" *
181     }
182     if$
183   }
184   if$
185 }

```

At the end of the reference, `fin.entry` writes out a final period and line break. We will handle the start of the reference a bit later.

```

186 FUNCTION {fin.entry}
187 { add.period$
188   write$
189   newline$
190 }

```

### 7.3 Convenience functions

We define some useful logical operations for use with `if$`.

```

192 FUNCTION {not}
193 { { #0 }
194   { #1 }
195   if$
196 }
197 FUNCTION {and}
198 { 'skip$
199   { pop$ #0 }
200   if$
201 }
202 FUNCTION {or}
203 { { pop$ #1 }
204   'skip$
205   if$
206 }

```

`new.block.checkb` consumes two tokens and starts a new block if at least one is non-empty.

```

207 FUNCTION {new.block.checkb}
208 { empty$
209   swap$ empty$

```

```

210 and
211   'skip$
212   'new.block
213   if$
214 }

```

field.or.null replaces an empty (null) field value with an empty string.

```

215 FUNCTION {field.or.null}
216 { duplicate$ empty$
217   { pop$ "" }
218   'skip$
219   if$
220 }

```

The emphasize function applies emphasis to the topmost token.

```

221 FUNCTION {emphasize}
222 { duplicate$ empty$
223   { pop$ "" }
224   { "\emph{" swap$ * "}" * }
225   if$
226 }

```

The tie.or.space.prefix function prefixes a string with a non-breaking space if it is up to three characters long, or a normal space otherwise.

```

227 FUNCTION {tie.or.space.prefix}
228 { duplicate$ text.length$ #3 <
229   { "~" }
230   { " " }
231   if$
232   swap$
233 }

```

The capitalize function makes putting strings into sentence case a little more robust.

```

234 FUNCTION {capitalize}
235 { "u" change.case$ "t" change.case$ }

```

The space.word function adds a space to either side of a string.

```

236 FUNCTION {space.word}
237 { " " swap$ * " " * }

```

either.or.check takes a string and a field, and if the field is not empty, writes a warning to the log file.

```

238 FUNCTION {either.or.check}
239 { empty$
240   'pop$
241   { "can't use both " swap$ * " fields in " * cite$ * warning$ }
242   if$
243 }

```

is.online checks whether the entry should have '[Online]' in it somewhere.

```

244 FUNCTION {is.online}
245 { online.shown
246   { #0 }
247   { url empty$
248     doi empty$ and
249     urldate empty$ and
250     urlyear empty$ and
251     { type$ "online" =
252       { #1 }
253       { #0 }
254       if$
255     }
256     { #1 }

```

```
257     if$
258   }
259   if$
260 }
```

## 7.4 Bibliography strings

Instead of dropping literal English words into the text of references, we will use these functions instead. This makes it easier to translate the style into other languages.

```
261 FUNCTION {bbl.and}
262 { "and" }
263
264 FUNCTION {bbl.byeditor}
265 { "Ed. by" }
266
267 FUNCTION {bbl.etal}
268 { "et~al." }
269
270 FUNCTION {bbl.editors}
271 { "eds" }
272
273 FUNCTION {bbl.editor}
274 { "ed." }
275
276 FUNCTION {bbl.edby}
277 { "edited by" }
278
279 FUNCTION {bbl.translator}
280 { "Trans." }
281
282 FUNCTION {bbl.edition}
283 { "ed." }
284
285 FUNCTION {bbl.volume}
286 { "vol." }
287
288 FUNCTION {bbl.of}
289 { "of" }
290
291 FUNCTION {bbl.number}
292 { "no." }
293
294 FUNCTION {bbl.nr}
295 { "no." }
296
297 FUNCTION {bbl.in}
298 { "in" }
299
300 FUNCTION {bbl.pages}
301 { "pp." }
302
303 FUNCTION {bbl.page}
304 { "p." }
305
306 FUNCTION {bbl.chapter}
307 { "chap." }
308
309 FUNCTION {bbl.techrep}
310 { "Tech. Rep." }
311
312 FUNCTION {bbl.mthesis}
313 { "Master's thesis" }
314
315 FUNCTION {bbl.phdthesis}
316 { "Ph.D. thesis" }
317
318 FUNCTION {bbl.first}
319 { "1st" }
320
321 FUNCTION {bbl.second}
```



```

322 { "2nd" }
323
324 FUNCTION {bbl.third}
325 { "3rd" }
326
327 FUNCTION {bbl.fourth}
328 { "4th" }
329
330 FUNCTION {bbl.fifth}
331 { "5th" }
332
333 FUNCTION {bbl.st}
334 { "st" }
335
336 FUNCTION {bbl.nd}
337 { "nd" }
338
339 FUNCTION {bbl.rd}
340 { "rd" }
341
342 FUNCTION {bbl.th}
343 { "th" }
344
345 FUNCTION {bbl.commission}
346 { "Commission Decision" }
347
348 FUNCTION {bbl.online}
349 { "[Online]" }
350
351 FUNCTION {bbl.nodate}
352 { "n.d." }
353
354 FUNCTION {bbl.inpreparation}
355 { "preprint" }
356
357 FUNCTION {bbl.submitted}
358 { "preprint" }
359
360 FUNCTION {bbl.inpress}
361 { "in press" }
362
363 FUNCTION {bbl.unpublished}
364 { "Unpublished" }

```

These macros are codes that users can put in their .bib file instead of literal strings.

```

366 MACRO {jan} {"January"}
367
368 MACRO {feb} {"February"}
369
370 MACRO {mar} {"March"}
371
372 MACRO {apr} {"April"}
373
374 MACRO {may} {"May"}
375
376 MACRO {jun} {"June"}
377
378 MACRO {jul} {"July"}
379
380 MACRO {aug} {"August"}
381
382 MACRO {sep} {"September"}
383
384 MACRO {oct} {"October"}
385
386 MACRO {nov} {"November"}
387
388 MACRO {dec} {"December"}

```

This function takes a cardinal number and replaces it with the corresponding English ordinal.

```

390 FUNCTION {eng.ord}
391 { duplicate$ "1" swap$ *
392 #~2 #1 substring$ "1" =
393   { bbl.th * }
394   { duplicate$ #-1 #1 substring$
395     duplicate$ "1" =
396       { pop$ bbl.st * }
397       { duplicate$ "2" =
398         { pop$ bbl.nd * }
399         { "3" =
400           { bbl.rd * }
401           { bbl.th * }
402           if$
403         }
404         if$
405       }
406     if$
407   }
408   if$
409 }

```

word.in provides the 'In:' before book titles.

```

410 FUNCTION {word.in}
411 { bbl.in capitalize
412 ":" *
413 " " * }

```

## 7.5 Writing out citation data to a file

format.names consumes two tokens (field, string field name) and returns a list of names in the form 'Surname, Suffix, Initials Prefix'. To help with this, we define some additional integers and strings.

```

415 INTEGERS { nameptr namesleft numnames }
416 STRINGS { bibinfo }
417 FUNCTION {format.names}
418 { 'bibinfo :=
419   duplicate$ empty$
420   'skip$
421   { 's :=
422     "" 't :=
423     #1 'nameptr :=
424     s num.names$ 'numnames :=
425     numnames 'namesleft :=
426     { namesleft #0 > }
427     { s nameptr
428       "{11}{, jj}{, f{.}.}{~vv}"
429       format.name$
430       bibinfo bibinfo.check
431       't :=
432       nameptr #1 >
433       { namesleft #1 >
434         { ", " * t * }
435         { s nameptr "{11}" format.name$ duplicate$ "others" =
436           { 't := }
437           { pop$ }
438           if$
439           t "others" =
440           { " " * bbl.etal *
441             }
442           { bbl.and
443             space.word * t *
444           }
445           if$
446         }
447       }
448     }
449     't
450     if$
451     nameptr #1 + 'nameptr :=

```

```

452     namesleft #1 - 'namesleft :=
453     }
454     while$
455     }
456     if$
457 }

```

`format.names.ed` consumes two tokens (field, string field name) and writes out a list of names in the form 'Initials Prefix Surname Suffix'.

```

458 FUNCTION {format.names.ed}
459 { 'bibinfo :=
460   duplicate$ empty$
461   'skip$
462   { 's :=
463     "" 't :=
464     #1 'nameptr :=
465     s num.names$ 'numnames :=
466     numnames 'namesleft :=
467     { namesleft #0 > }
468     { s nameptr
469       "{f{.}.~}{vv~}{ll}{ jj}"
470       format.name$
471       bibinfo bibinfo.check
472       't :=
473       nameptr #1 >
474       { namesleft #1 >
475         { ", " * t * }
476         { s nameptr "{ll}" format.name$ duplicate$ "others" =
477           { 't := }
478           { pop$ }
479           if$
480           t "others" =
481             { " " * bbl.etal *
482             }
483             { bbl.and
484               space.word * t *
485             }
486           if$
487         }
488         if$
489       }
490       't
491       if$
492       nameptr #1 + 'nameptr :=
493       namesleft #1 - 'namesleft :=
494     }
495     while$
496   }
497   if$
498 }

```

`format.authors` returns author names.

```

499 FUNCTION {format.authors}
500 { author "author" format.names
501 }

```

`format.byeditors` returns editor names, and prepended by 'Ed. by'.

```

502 FUNCTION {format.byeditors}
503 { editor "editor" format.names.ed duplicate$ empty$ 'skip$
504   { bbl.byeditor
505     " " *
506     swap$ *
507   }
508   if$
509 }

```

`format.editors` returns editor names, and appends 'ed.' or 'eds' as appropriate.

```

510 FUNCTION {get.bbl.editor}
511 { editor num.names$ #1 > 'bbl.editors 'bbl.editor if$ }
512 FUNCTION {format.editors}
513 { editor "editor" format.names duplicate$ empty$ 'skip$
514   { "," *
515     " " *
516     get.bbl.editor
517     *
518   }
519   if$
520 }

```

format.translators does likewise for translators, in the form '(name, Trans.)'.

```

521 FUNCTION {format.translators}
522 { translator "translator" format.names.ed duplicate$ empty$ 'skip$
523   { "," *
524     " " *
525     bbl.translator *
526     "(" swap$ *
527     ")" *
528   }
529   if$
530 }

```

format.full.names consumes one token (a name field) and, contrary to what the name might suggest, returns a list of names in the form 'Prefix Surname'. It is called 'full' because the list is not truncated.

```

531 FUNCTION {format.full.names}
532 {'s :=
533 "" 't :=
534 #1 'nameptr :=
535 s num.names$ 'numnames :=
536 numnames 'namesleft :=
537 { namesleft #0 > }
538 { s nameptr
539   "{vv-}{ll}" format.name$
540   't :=
541   nameptr #1 >
542   { namesleft #1 >
543     { " " * t * }
544     { s nameptr "{ll}" format.name$ duplicate$ "others" =
545       { 't := }
546       { pop$ }
547       if$
548       t "others" =
549       { " " * bbl.etal *
550       }
551       { bbl.and
552         space.word * t *
553       }
554       if$
555     }
556     if$
557   }
558   't
559   if$
560   nameptr #1 + 'nameptr :=
561   namesleft #1 - 'namesleft :=
562 }
563 while$
564 }

```

author.editor.key.full returns the author part of an author-year citation. In order of preference, it outputs the author surnames, editor surnames, the key field, or the first three characters of the entry key. It is used in book and inbook entry types.

```

565 FUNCTION {author.editor.key.full}
566 { author empty$
567   { editor empty$

```

```

568     { key empty$
569       { cite$ #1 #3 substring$ }
570       'key
571       if$
572     }
573     { editor format.full.names }
574     if$
575   }
576   { author format.full.names }
577   if$
578 }

```

editor.key.full does similar but skips the author names. It is used for the proceedings entry type.

```

579 FUNCTION {editor.key.full}
580 { editor empty$
581   { key empty$
582     { cite$ #1 #3 substring$ }
583     'key
584     if$
585   }
586   { editor format.full.names }
587   if$
588 }

```

author.key.full does similar but skips the editor names. It is used for all other entry types.

```

589 FUNCTION {author.key.full}
590 { author empty$
591   { key empty$
592     { cite$ #1 #3 substring$ }
593     'key
594     if$
595   }
596   { author format.full.names }
597   if$
598 }

```

make.full.names chooses between the above.

```

599 FUNCTION {make.full.names}
600 { type$ "book" =
601   type$ "inbook" =
602   or
603   'author.editor.key.full
604   { type$ "proceedings" =
605     'editor.key.full
606     'author.key.full
607     if$
608   }
609   if$
610 }

```

output.bibitem writes out the start of an entry in the bibliography, which looks like \bibitem[`{author-year elements}`]{`entry key`}. The author-year part uses special natbib syntax: either 'Author(Year)' or 'Truncated author list(Year)Full author list'. The calc.label function below supplies the truncated author list, opening parenthesis and year; you see here the closing parenthesis and conditional inclusion of the full list. The function then starts a new line, puts the entry driver into the before.all output state, and puts an empty string at the top of the stack.

```

611 FUNCTION {output.bibitem}
612 { newline$
613   "\bibitem[{" write$
614   label write$
615   "]" make.full.names duplicate$ short.list =
616   { pop$ }
617   { * }
618   if$
619   "][{" * write$

```

```

620 cite$ write$
621 "]" write$
622 newline$
623 ""
624 #0 'online.shown :=
625 before.all 'output.state :=
626 }

```

`format.key` takes a field (author or editor). If the field is empty, it replaces it with the key field (if not non-null). Otherwise, replaces it with an empty string. It is used as an extra defence against an entry beginning with the date.

```

627 FUNCTION {format.key}
628 { empty$
629   { key field.or.null }
630   { "" }
631   if$
632   nameaddon duplicate$ empty$
633   'pop$
634   { swap$ duplicate$ empty$
635     'skip$
636     { " " * }
637     if$
638     swap$ "[" swap$ * * "]" *
639     continue.clause
640   }
641   if$
642 }

```

## 7.6 Field formats

Each of the `format.*` functions below (and above) add a single string (possibly empty) to the stack, based on the value of one or more fields.

### 7.6.1 Title block

`select.language` checks if the string at the top of the stack is empty, and if not, wraps `{\selectlanguage{language}...}` around it.

```

643 FUNCTION {select.language}
644 { duplicate$ empty$
645   'pop$
646   { language empty$
647     'skip$
648     { "\selectlanguage{" language * "}" * swap$ * "}" * }
649     if$
650   }
651   if$
652 }

```

`format.title` returns the title and language fields, where the title should be in the upright shape.

```

653 FUNCTION {format.title}
654 { title
655   duplicate$ empty$ 'skip$
656   { "t" change.case$ }
657   if$
658   "title" bibinfo.check
659   duplicate$ empty$
660   'skip$
661   { select.language }
662   if$
663 }

```

`format.btitle` returns the title and language fields, where the title should be in the italic shape.

```

664 FUNCTION {format.btitle}
665 { title
666   duplicate$ empty$ 'skip$
667   { "t" change.case$ }
668   if$
669   "title" bibinfo.check
670   duplicate$ empty$
671   'skip$
672   { emphasize select.language }
673   if$
674 }

```

`format.avtitle` returns the title and language fields, where the shape of the title (upright or italic) depends on the type.

```

675 FUNCTION {format.avtitle}
676 { title
677   duplicate$ empty$ 'skip$
678   { "t" change.case$ }
679   if$
680   "title" bibinfo.check
681   duplicate$ empty$
682   'skip$
683   { type duplicate$ empty$
684     { pop$ emphasize }
685     { duplicate$ "TV" = swap$ "Radio" = or
686       'skip$
687       'emphasize
688       if$
689     }
690     if$
691     select.language
692   }
693   if$
694 }

```

`format.titleaddon` returns the `titleaddon` field. This is the `bath.bst` version:

```

695 FUNCTION {format.titleaddon}
696 { titleaddon
697   duplicate$ empty$ 'skip$
698   { "titleaddon" bibinfo.check }
699   if$
700 }

```

This is the `bathx.bst` version:

```

695 FUNCTION {format.titleaddon}
696 { titleaddon
697   duplicate$ empty$ 'skip$
698   { "[" swap$ "titleaddon" bibinfo.check * "]" * }
699   if$
700 }

```

`format.version` returns the version field.

```

701 FUNCTION {format.version}
702 { version
703   duplicate$ empty$ 'skip$
704   { "(v." swap$ *
705     "version" bibinfo.check
706     ")" *
707   }
708   if$
709 }

```

`print.labeltitle.addenda` is a convenience for writing out the elements that normally follow the title when it appears at the head of the reference.

```

710 FUNCTION {print.labeltitle.addenda}
711 { continue.clause

```

In `bath.bst` it prints the version and then `titleaddon` field:

```
712 online.shown
713 'skip$
714 { format.version output
715   continue.clause
716   format.titleaddon output
717   #1 'online.shown :=
718 }
719 if$
```

In `bathx.bst` it prints the `titleaddon` field, then the version, then either the `entrysubtype` or the `bb1.online` string as appropriate:

```
712 format.titleaddon output
713 continue.clause
714 type$ duplicate$ "legislation" = not
715 swap$ "jurisdiction" = not
716 and
717 { format.version output
718   is.online
719   { entrysubtype duplicate$ empty$
720     { pop$
721       journal empty$
722       booktitle empty$
723         type$ "inreference" =
724         author empty$
725         and
726       or
727     and
728     library empty$
729     type$ duplicate$ "image" =
730     swap$ duplicate$ "audio" =
731     swap$ duplicate$ "video" =
732     swap$ duplicate$ "music" =
733     swap$ "movie" =
734     or
735     or
736     or
737     or
738     or
739     and
740     series empty$
741     type field.or.null
742     duplicate$ "TV" =
743     swap$ "Radio" =
744     or not
745     or
746     and
747     { continue.clause
748       bbl.online output
749       #1 'online.shown :=
750     }
751     'skip$
752     if$
753   }
754   { "entrysubtype" bibinfo.check
755     "[" swap$ * "]" *
756     continue.clause output
757   }
758   if$
759 }
760 'skip$
761 if$
762 }
763 'skip$
764 if$
```

`print.title.addenda` is a convenience for writing out the elements that normally follow the title in its regular position. It is similar to `print.labeltitle.addenda` but has somewhat different logic.



```

765 }
766 FUNCTION {print.title.addenda}
767 { continue.clause

```

In bath.bst it prints the version and then titleaddon field:

```

723 online.shown
724   'skip$
725   { format.version output
726     continue.clause
727     format.titleaddon output
728     #1 'online.shown :=
729   }
730   if$

```

In bathx.bst it prints the titleaddon field, then the version, then either the entrysubtype or the bbl.online string as appropriate:

```

768 type$ duplicate$ "legislation" = not
769 swap$ "jurisdiction" = not
770 and
771   { format.titleaddon output
772     continue.clause
773   }
774   'skip$
775   if$
776   format.version output
777   is.online
778   { entrysubtype duplicate$ empty$
779     type$ duplicate$ "legislation" =
780     swap$ duplicate$ "jurisdiction" =
781     swap$ "legal" =
782     or
783     or
784     or
785     { pop$
786       journal empty$
787       booktitle empty$
788       type$ "inreference" =
789       author empty$
790       and
791       or
792       and
793       library empty$
794       type$ duplicate$ "image" =
795       swap$ duplicate$ "audio" =
796       swap$ duplicate$ "video" =
797       swap$ duplicate$ "music" =
798       swap$ "movie" =
799       or
800       or
801       or
802       or
803       or
804       and
805       series empty$
806       type field.or.null
807       duplicate$ "TV" =
808       swap$ "Radio" =
809       or not
810       or
811       and
812       { continue.clause
813         bbl.online output
814         #1 'online.shown :=
815       }
816       'skip$
817       if$
818     }
819     { "entrysubtype" bibinfo.check
820       "[" swap$ * "]" *
821       continue.clause output

```

```

822     }
823     if$
824   }
825   'skip$
826 if$

```

We finish off with editor/translator information.

```

827 continue.clause
828 type$ "reference" =
829   { format.byeditors output }
830   'skip$
831 if$
832 format.translators output
833 }

```

`format.label.booktitle` returns the booktitle field upright

```

834 FUNCTION {format.label.booktitle}
835 { booktitle
836   duplicate$ empty$ 'skip$
837   { "t" change.case$ }
838   if$
839   "booktitle" bibinfo.check
840 }

```

`format.booktitle` returns the booktitle field in italics.

```

841 FUNCTION {format.booktitle}
842 { format.label.booktitle
843   emphasize
844 }

```

`format.in.ed.booktitle` returns the booktitle, prepended by ‘In: editors, eds.’

```

845 FUNCTION {format.in.ed.booktitle}
846 { format.booktitle
847   duplicate$ empty$
848   'skip$
849   {

```

In `bathx.bst`, we conditionally include the `bbl.online` string:

```

850   is.online
851   { " " *
852     bbl.online *
853     #1 'online.shown :=
854   }
855   'skip$
856 if$

```

```

857   editor "editor" format.names.ed duplicate$ empty$ 'pop$
858   { "," *
859     " " *
860     get.bbl.editor add.period$
861     " " *
862     * swap$
863     *
864     word.in swap$ *
865   }
866   if$
867 }
868 if$
869 }

```

`legal.journal.shape` emphasizes the preceding string if it matches ‘OJ’ but otherwise does nothing.

```

870 FUNCTION {legal.journal.shape}
871 { duplicate$ "OJ" =

```

```

872     'emphasize
873     'skip$
874     if$
875 }

```

## 7.6.2 Dates

In `bath.bst`, `nodate.check` doesn't do anything:

```

775 FUNCTION {nodate.check}
776 {
777     skip$
778 }

```

In `bathx.bst`, it looks at the most recent token. If it is empty, replaces it with 'n.d.' unless either `sortyear` or the `nonodate` option is present.

```

876 FUNCTION {nodate.check}
877 {
878     duplicate$ empty$
879     sortyear empty$
880     and
881     { options field.or.null "nonodate" =
882       'skip$
883       { pop$ bbl.nodate }
884       if$
885     }
886     'skip$
887     if$
888 }

```

`format.date` returns the year and any disambiguation label.

```

889 FUNCTION {format.date}
890 { year "year" bibinfo.check nodate.check
891   extra.label *
892 }

```

`format.jur.date` wraps the year and any disambiguation label in parentheses if a volume number is provided or if `year-essential=false` has been given as the options. If there is no volume number, or if `year-essential=true` has been given as the options, then the year will be wrapped in square brackets or, if `scottish-style` has been given as the options or `sc` as the keywords, left bare. But if the journal is 'OJ' or 'ECR', the date is wrapped in brackets regardless. Not complicated at all, really.

```

893 FUNCTION {format.date.brackets}
894 { format.date
895   "[" swap$ * "]" *
896 }
897 FUNCTION {format.date.parens}
898 { format.date
899   "(" swap$ * ")" *
900 }
901 FUNCTION {format.jur.date.essential}
902 { options field.or.null "scottish-style" =
903   keywords field.or.null "sc" =
904   or
905   { end.clause format.date }
906   'format.date.brackets
907   if$
908 }
909 FUNCTION {format.jur.date}
910 { institution empty$
911   number empty$
912   or
913   { journal field.or.null 's :=
914     s "OJ" =
915     s "ECR" =
916     or

```

```

917     'format.date.brackets
918     { options field.or.null "year-essential=false" =
919       'format.date.parens
920       { volume empty$
921         'format.jur.date.essential
922         { options field.or.null "year-essential=true" =
923           'format.jur.date.essential
924           'format.date.parens
925             if$
926           }
927         if$
928       }
929     if$
930   }
931 if$
932 }
933 { end.clause format.date }
934 if$
935 }

```

`format.eu.date` wraps the year and any disambiguation label in square brackets if (and only if) the journal title is 'OJ' or 'ECR'.

```

936 FUNCTION {format.eu.date}
937 { journal field.or.null 's :=
938   s "OJ" =
939   s "ECR" =
940   or
941     'format.date.brackets
942     'format.date
943   if$
944 }

```

### 7.6.3 Series, types and numbering

`format.series` returns the case-altered series field.

```

945 FUNCTION {format.series}
946 { series
947   duplicate$ empty$ 'skip$
948   { "t" change.case$ }
949   if$
950   "series" bibinfo.check
951 }

```

`format.bvolume` returns the volume and series information (in italics), with a comma between. A warning is given if `number` is also given.

```

952 FUNCTION {format.bvolume}
953 { volume empty$
954   { "" }
955   { bbl.volume volume tie.or.space.prefix
956     "volume" bibinfo.check * *
957     format.series
958     duplicate$ empty$ 'pop$
959     { emphasize ", " * swap$ * }
960     if$
961     "volume and number" number either.or.check
962   }
963   if$
964 }

```

`format.number.series` returns an empty string if a volume has been specified. Otherwise, it returns the series and the number separated by a space.

```

965 FUNCTION {format.number.series}
966 { volume empty$
967   { number empty$
968     { series field.or.null }
969     { series empty$

```

```

970     { number "number" bibinfo.check }
971     { format.series
972       number tie.or.space.prefix "number" bibinfo.check * *
973     }
974     if$
975   }
976   if$
977 }
978 { "" }
979 if$
980 }

```

`format.series.episode` returns the case-altered series and the number, separated by a space and both emphasised.

```

981 FUNCTION {format.series.episode}
982 { series
983   duplicate$ empty$
984   'skip$
985   { "t" change.case$ }
986   if$
987   "series" bibinfo.check emphasize
988   number "number" bibinfo.check emphasize
989   duplicate$ empty$
990   { * }
991   { swap$
992     duplicate$ empty$
993     { * }
994     { ", " * swap$ * }
995     if$
996   }
997   if$
998 }

```

`is.num` converts a string to an integer and ensures it is in the range 0–9.

```

999 FUNCTION {is.num}
1000 { chr.to.int$
1001   duplicate$ "0" chr.to.int$ < not
1002   swap$ "9" chr.to.int$ > not and
1003 }

```

`extract.num` takes a string. If it begins with an integer, it returns just that integer. Otherwise it returns the whole string.

```

1004 FUNCTION {extract.num}
1005 { duplicate$ 't :=
1006   "" 's :=
1007   { t empty$ not }
1008   { t #1 #1 substring$
1009     t #2 global.max$ substring$ 't :=
1010     duplicate$ is.num
1011     { s swap$ * 's := }
1012     { pop$ "" 't := }
1013     if$
1014   }
1015   while$
1016   s empty$
1017   'skip$
1018   { pop$ s }
1019   if$
1020 }

```

`convert.edition` should operate on the edition field. If the value begins with a number (or is an English ordinal word from ‘first’ to ‘fifth’), then it returns the bibliography string version of the ordinal number (1st, 2nd, 3rd...). Otherwise, returns the edition field verbatim.

```

1021 FUNCTION {convert.edition}
1022 { extract.num "1" change.case$ 's :=
1023   s "first" = s "1" = or

```

```

1024 { bbl.first 't := }
1025 { s "second" = s "2" = or
1026   { bbl.second 't := }
1027   { s "third" = s "3" = or
1028     { bbl.third 't := }
1029     { s "fourth" = s "4" = or
1030       { bbl.fourth 't := }
1031       { s "fifth" = s "5" = or
1032         { bbl.fifth 't := }
1033         { s #1 #1 substring$ is.num
1034           { s eng.ord 't := }
1035           { edition 't := }
1036           if$
1037         }
1038       if$
1039     }
1040   if$
1041 }
1042 if$
1043 }
1044 if$
1045 }
1046 if$
1047 t
1048 }

```

`format.edition` returns the converted edition field and ‘ed.’

```

1049 FUNCTION {format.edition}
1050 { edition duplicate$ empty$ 'skip$
1051   { convert.edition
1052     output.state mid.sentence =
1053     { "1" }
1054     { "t" }
1055     if$ change.case$
1056     "edition" bibinfo.check
1057     " " * bbl.edition *
1058   }
1059 if$
1060 }

```

If the type is provided, `format.thesis.type` deletes the preceding string (the default value) and writes the type field with an initial capital. Otherwise it does nothing.

```

1061 FUNCTION {format.thesis.type}
1062 { type duplicate$ empty$
1063   'pop$
1064   { swap$ pop$
1065     "t" change.case$ "type" bibinfo.check
1066   }
1067 if$
1068 }

```

`format.tr.number` returns the series, type and number fields as one of ‘(number)’, ‘(type number)’, ‘(series, number)’, or ‘(series, type number)’. If a series is given, a `continue clause` is inserted to remove the preceding comma.

If the series begins ‘HL<sub>□</sub>’ (with or without braces, but it should be with), it is assumed to be a House of Lords paper, thus the number acquires additional parentheses. If it equals ‘C’, ‘Cd’, ‘Cmd’, ‘Cmnd’ or ‘Cm’, it is followed by a dot rather than a comma (implemented as `add.cmd.number`). To help with this, `format.legr.number` consumes one token (the number) and returns the series and number as one token.

```

1069 FUNCTION {add.cmd.number}
1070 { "." swap$ "number" bibinfo.check *
1071 }
1072 FUNCTION {format.legr.number}
1073 { series field.or.null 's :=
1074   s "C" =
1075     'add.cmd.number
1076   { s "Cd" =

```

```

1077 'add.cmd.number
1078 { s "Cmd" =
1079 'add.cmd.number
1080 { s "Cmnd" =
1081 'add.cmd.number
1082 { s "Cm" =
1083 'add.cmd.number
1084 { s purify$ #1 #3 substring$ 't :=
1085 t "HL " =
1086 { ", (" swap$
1087 "number" bibinfo.check *
1088 ")" *
1089 }
1090 { ", " swap$
1091 "number" bibinfo.check *
1092 }
1093 if$
1094 }
1095 if$
1096 }
1097 if$
1098 }
1099 if$
1100 }
1101 if$
1102 }
1103 if$
1104 }
1105 FUNCTION {format.tr.number}
1106 { series duplicate$ empty$
1107 { pop$
1108 type duplicate$ empty$
1109 { pop$
1110 number duplicate$ empty$
1111 { pop$ ""
1112 }
1113 { "(" swap$
1114 "number" bibinfo.check
1115 * ")" *
1116 }
1117 if$
1118 }
1119 { "(" swap$
1120 "t" change.case$ "type" bibinfo.check
1121 *
1122 number duplicate$ empty$
1123 'pop$
1124 { tie.or.space.prefix
1125 "number" bibinfo.check
1126 * *
1127 }
1128 if$
1129 ")" *
1130 }
1131 if$
1132 }
1133 { "t" change.case$ "(" swap$ *
1134 type duplicate$ empty$
1135 { pop$
1136 number duplicate$ empty$
1137 'pop$
1138 { format.legr.number * }
1139 if$
1140 }
1141 { "t" change.case$ "type" bibinfo.check
1142 *
1143 number duplicate$ empty$
1144 'pop$
1145 { tie.or.space.prefix
1146 "number" bibinfo.check
1147 * *
1148 }
1149 if$

```

```

1150     }
1151     if$
1152     ")" *
1153     continue.clause
1154     }
1155 if$
1156 }

```

`format.manual.number` returns the series, type and number fields as one of ‘number’, ‘type number’ ‘series, number’, or ‘series, number. type’. The last of these formulations is for compatibility with previous versions, but may change in future.

```

1157 FUNCTION {format.manual.number}
1158 { series duplicate$ empty$
1159   { pop$
1160     type duplicate$ empty$
1161     { pop$
1162       number duplicate$ empty$
1163       { pop$ ""
1164       }
1165       { "number" bibinfo.check
1166       }
1167     if$
1168     }
1169     { "t" change.case$ "type" bibinfo.check
1170     number duplicate$ empty$
1171     'pop$
1172     { tie.or.space.prefix
1173       "number" bibinfo.check
1174       * *
1175     }
1176     if$
1177     }
1178   if$
1179 }
1180 { "t" change.case$
1181 number duplicate$ empty$
1182 'pop$
1183 { "number" bibinfo.check
1184   ", " swap$ * *
1185 }
1186 if$
1187 type duplicate$ empty$
1188 'pop$
1189 { ". " swap$ *
1190   "t" change.case$ "type" bibinfo.check
1191   *
1192 }
1193 if$
1194 }
1195 if$
1196 }

```

`format.chapter` returns the chapter field in the form ‘c.1’, for use with UK primary legislation.

```

1197 FUNCTION {format.chapter}
1198 { chapter duplicate$ empty$
1199   { pop$ "" }
1200   { "chapter" bibinfo.check
1201     "c." swap$ *
1202   }
1203 if$
1204 }

```

`format.series.number.chapter` returns the series and chapter/number fields in the form ‘(series, c.1)’ or ‘(series, number)’, for use with UK legislation.

```

1205 FUNCTION {format.series.number.chapter}
1206 { series duplicate$ empty$
1207   { pop$ "(" }
1208   { "series" bibinfo.check

```



```

1209     "(" swap$ *
1210   }
1211   if$
1212   type duplicate$ empty$
1213   { pop$
1214     number duplicate$ empty$
1215     'pop$
1216     { "number" bibinfo.check
1217       swap$ duplicate$ "(" =
1218         { swap$ * }
1219         { ", " * swap$ * }
1220       if$
1221     }
1222     if$
1223     chapter duplicate$ empty$
1224     'pop$
1225     { "chapter" bibinfo.check
1226       swap$ duplicate$ "(" =
1227         { swap$ * }
1228         { ", c." * swap$ * }
1229       if$
1230     }
1231     if$
1232   }
1233   { "type" bibinfo.check
1234     swap$ duplicate$ "(" =
1235       { swap$ * }
1236       { ", " * swap$ * }
1237     if$
1238     number duplicate$ empty$
1239     { pop$
1240       chapter duplicate$ empty$
1241       'pop$
1242       { "chapter" bibinfo.check
1243         " c." swap$ * *
1244       }
1245       if$
1246     }
1247     { "number" bibinfo.check
1248       " " swap$ * *
1249     chapter duplicate$ empty$
1250     'pop$
1251     { "chapter" bibinfo.check
1252       ", c." swap$ * *
1253     }
1254     if$
1255   }
1256   if$
1257 }
1258 if$
1259 ")" *
1260 duplicate$ "(" =
1261   { pop$ "" }
1262   'skip$
1263   if$
1264 }

```

format.case.number adds to the stack '(number or casenumber)' if either of those fields is provided with no institution; or '(userb or casenumber) Commission Decision number' if institution is 'Commission'; otherwise adds an empty string.

```

1265 FUNCTION {format.case.number}
1266 { number field.or.null 's :=
1267   s empty$
1268     { casenumber field.or.null 's := }
1269     'skip$
1270   if$
1271   s empty$
1272     { "" }
1273     { institution field.or.null duplicate$ empty$
1274       { pop$
1275         s "number" bibinfo.check

```

```

1276     "(" swap$ * )" *
1277   }
1278   { "Commission" =
1279     { userb field.or.null 's :=
1280       s empty$
1281         { casenumber field.or.null 's := }
1282         'skip$
1283         if$
1284         s duplicate$ empty$
1285           { "" }
1286           { "(" swap$ * )" * " " }
1287         if$
1288         number duplicate$ empty$
1289           { pop$ pop$ }
1290           { bbl.commission " " *
1291             swap$ * * *
1292           }
1293         if$
1294       }
1295     { "" }
1296   if$
1297 }
1298 if$
1299 }
1300 if$
1301 }

```

`case.check` checks if the entry should be punctuation with a full stop.

```

1302 FUNCTION {case.check}
1303 { institution duplicate$ empty$
1304   { pop$ continue.clause }
1305   { "Commission" =
1306     'continue.clause
1307     'new.block
1308     if$
1309   }
1310 if$
1311 }

```

`eu.case.check` checks if the case is in OJ or has a number, in which case punctuation is suppressed.

```

1312 FUNCTION {eu.case.check}
1313 { journal field.or.null "OJ" =
1314   number empty$ not
1315   or
1316   casenumber empty$ not
1317   or
1318   'continue.clause
1319   'new.block
1320 if$
1321 }

```

#### 7.6.4 Notes

`format.note` returns the note field, but changes the case of the first letter according to the output state (unless protected by braces).

```

1322 FUNCTION {format.note}
1323 { note empty$
1324   { "" }
1325   { note #1 #1 substring$
1326     duplicate$ "{" =
1327     'skip$
1328     { output.state mid.sentence =
1329       output.state mid.clause =
1330       or
1331       { "l" }
1332       { "u" }
1333     if$

```

```

1334     change.case$
1335   }
1336   if$
1337   note #2 global.max$ substring$ * "note" bibinfo.check
1338 }
1339 if$
1340 }

```

### 7.6.5 Publication block

`prepend.address` prepends the address field to the string at the top of the stack, which should be a publisher (or equivalent) bibinfo string. If both the input string and address field are non-empty, a colon and space are interposed between them.

```

1341 FUNCTION {prepend.address}
1342 { 't :=
1343   ""
1344   address empty$ t empty$ and
1345   'skip$
1346   { address "address" bibinfo.check *
1347     t empty$
1348     'skip$
1349     { address empty$
1350       'skip$
1351       { ":" * }
1352       if$
1353       t *
1354     }
1355     if$
1356   }
1357   if$
1358 }

```

The next three functions return the address plus the publisher, organization and institution respectively.

```

1359 FUNCTION {format.publisher.address}
1360 { publisher "publisher" bibinfo.warn prepend.address
1361 }
1362 FUNCTION {format.organization.address}
1363 { publisher empty$
1364   { organization "organization" bibinfo.check prepend.address }
1365   { publisher "publisher" bibinfo.warn prepend.address
1366     organization duplicate$ empty$
1367     'pop$
1368     { "organization" bibinfo.check ". " * swap$ * }
1369     if$
1370   }
1371   if$
1372 }
1373 FUNCTION {format.institution.address}
1374 { publisher empty$
1375   { institution "institution" bibinfo.check prepend.address }
1376   { publisher "publisher" bibinfo.warn prepend.address
1377     institution duplicate$ empty$
1378     'pop$
1379     { "institution" bibinfo.check ". " * swap$ * }
1380     if$
1381   }
1382   if$
1383 }

```

We add a macro for printing the library (repository) holding an eprint.

```

1384 FUNCTION {format.library}
1385 { library "library" bibinfo.check
1386   emphasize

```

In `bathx.bst`, we conditionally include the `bbl.online` string:

```

1387 duplicate$ empty$ not
1388 is.online
1389 and
1390 journal empty$
1391 and
1392 { " " * bbl.online *
1393 #1 'online.shown :=
1394 }
1395 'skip$
1396 if$

```

The library (or institution) field is also used for where a physical image is kept. As we have only one address field, it will attach to whichever is present out of organization, publisher and library. In case there is more than one, it will attach to publisher in preference to library and library in preference to organization.

```

1397 }
1398 FUNCTION {format.img.library}
1399 { library "library" bibinfo.check
1400 }
1401 FUNCTION {format.img.library.address}
1402 { library duplicate$ empty$
1403   { pop$
1404     institution duplicate$ empty$
1405     'skip$
1406     { "institution" bibinfo.check }
1407     if$
1408   }
1409   { "library" bibinfo.check }
1410   if$
1411   duplicate$ empty$
1412   { address "address" bibinfo.check * }
1413   { address duplicate$ empty$
1414     'pop$
1415     { "address" bibinfo.check
1416       swap$ ", " * swap$ *
1417     }
1418     if$
1419   }
1420   if$
1421 }
1422 FUNCTION {format.pub.org.lib.address}
1423 { publisher duplicate$ empty$
1424   { pop$
1425     organization duplicate$ empty$
1426     { pop$
1427       format.img.library.address }
1428     { "organization" bibinfo.check
1429       library empty$
1430       { prepend.address }
1431       { ". " *
1432         format.img.library.address *
1433       }
1434       if$
1435     }
1436     if$
1437   }
1438   { "publisher" bibinfo.warn prepend.address
1439     library empty$
1440     'skip$
1441     { ". " *
1442       format.img.library *
1443     }
1444     if$
1445   }
1446   if$
1447 }

```

## 7.6.6 Pinpointing

`multi.page.check` should operate on the `pages` field. If the value is a range, that is, contains '-', ',', or '+', returns 1, otherwise returns 0.

```
1449 INTEGERS { multiresult }
1450 FUNCTION {multi.page.check}
1451 { 't :=
1452   #0 'multiresult :=
1453   { multiresult not
1454     t empty$ not
1455     and
1456   }
1457   { t #1 #1 substring$
1458     duplicate$ "-" =
1459     swap$ duplicate$ "," =
1460     swap$ "+" =
1461     or or
1462     { #1 'multiresult := }
1463     { t #2 global.max$ substring$ 't := }
1464     if$
1465   }
1466   while$
1467   multiresult
1468 }
```

`n.dashify` converts hyphens in page ranges into en dashes.

```
1469 FUNCTION {n.dashify}
1470 { 't :=
1471   ""
1472   { t empty$ not }
1473   { t #1 #1 substring$ "-" =
1474     { t #1 #2 substring$ "---" = not
1475       { "---" *
1476         t #2 global.max$ substring$ 't :=
1477       }
1478       { { t #1 #1 substring$ "-" = }
1479         { "-" *
1480           t #2 global.max$ substring$ 't :=
1481         }
1482         while$
1483       }
1484       if$
1485     }
1486     { t #1 #1 substring$ *
1487       t #2 global.max$ substring$ 't :=
1488     }
1489     if$
1490   }
1491   while$
1492 }
```

`format.pages` returns the `pages` fields, prefixed by 'p.' or 'pp.' as appropriate.

```
1493 FUNCTION {format.pages}
1494 { pages duplicate$ empty$ 'skip$
1495   { duplicate$ multi.page.check
1496     { bbl.pages swap$
1497       n.dashify
1498     }
1499     { bbl.page swap$
1500     }
1501     if$
1502     "pages" bibinfo.check
1503     *
1504   }
1505   if$
1506 }
```

`format.journal.pages` does similar, but it looks at the preceding string and, if both are non-empty, inserts a comma between it and the `pages` string.

```

1507 FUNCTION {format.journal.pages}
1508 { pages duplicate$ empty$ 'pop$
1509   { swap$ duplicate$ empty$
1510     { pop$ pop$ format.pages }
1511     { ", " *
1512       swap$
1513       n.dashify
1514       pages multi.page.check
1515       'bbl.pages
1516       'bbl.page
1517       if$
1518       swap$
1519       "pages" bibinfo.check
1520     * *
1521   }
1522   if$
1523 }
1524 if$
1525 }

```

`format.journal.eid` returns the EID, but it looks at the preceding string and, if both are non-empty, inserts a comma between it and the EID string. The EID is a code used by some online-only journals to identify an article within a volume or issue instead of a page range (since the page numbering resets for each article, not each volume/issue).

```

1526 FUNCTION {format.journal.eid}
1527 { eid "eid" bibinfo.check
1528   duplicate$ empty$ 'pop$
1529   { swap$ duplicate$ empty$ 'skip$
1530     { ", " *
1531     }
1532     if$
1533     swap$ *
1534   }
1535   if$
1536 }

```

`format.vol.num` returns volume and number in the form ‘volume(number)’. It writes a warning to the log if `number` is given instead of `volume`: if a journal does not organize itself into volumes, treat every issue as its own volume!

```

1537 FUNCTION {format.vol.num}
1538 { volume field.or.null duplicate$ empty$
1539   'skip$
1540   { "volume" bibinfo.check
1541   }
1542   if$
1543   number "number" bibinfo.check duplicate$ empty$
1544   'skip$
1545   { swap$ duplicate$ empty$
1546     { "there's a number but no volume in " cite$ * warning$ }
1547     'skip$
1548     if$
1549     swap$
1550     "(" swap$ * ")" *
1551   }
1552   if$
1553 }

```

`format.vol.num.pages` returns the journal pinpointing in the form ‘volume(number), eid/pages’. Since the Harvard (Bath) style places publication state information in the volume position, the `pubstate` field from `biblatex` is emulated here. The `eventyear` if present is used instead of month, in order to support parliamentary debates.

```

1554 FUNCTION {format.vol.num.pages}
1555 { pubstate field.or.null
1556   duplicate$ "inpress" =
1557   { pop$ bbl.inpress }
1558   { duplicate$ "submitted" =
1559     { pop$ bbl.submitted }

```

```

1560     { duplicate$ "inpreparation" =
1561       { pop$ bbl.inpreparation }
1562       { pop$ "" }
1563       if$
1564     }
1565   if$
1566 }
1567 if$
1568 duplicate$ empty$
1569 { pop$
1570   format.vol.num *
1571   eventyear field.or.null duplicate$ empty$
1572   { pop$
1573     month "month" bibinfo.check duplicate$ empty$
1574     'pop$
1575     { swap$ duplicate$ empty$
1576       'pop$
1577       { ", " * swap$ * }
1578       if$
1579     }
1580     if$
1581   }
1582   { swap$ duplicate$ empty$
1583     'pop$
1584     { ", " * swap$ * }
1585     if$
1586   }
1587   if$
1588   eid empty$
1589   { format.journal.pages }
1590   { format.journal.eid }
1591   if$
1592 }
1593 'skip$
1594 if$
1595 }

```

format.chapter.pages returns pinpointing information for books, in the form 'chap. chapter, p.pages'.

```

1596 FUNCTION {format.chapter.pages}
1597 { chapter empty$
1598   'format.pages
1599   { type empty$
1600     { bbl.chapter }
1601     { type "1" change.case$
1602       "type" bibinfo.check
1603     }
1604     if$
1605     chapter tie.or.space.prefix
1606     "chapter" bibinfo.check
1607     * *
1608     pages empty$
1609     'skip$
1610     { ", " * format.pages * }
1611     if$
1612   }
1613   if$
1614 }

```

format.journal.series.vol.pages returns the journal title and pinpointing in the form 'volume journal pages', 'journal volume-pages' or 'journal series volume/pages', depending on the journal.

```

1616 FUNCTION {format.journal.series.vol.pages}
1617 { journal field.or.null
1618   duplicate$ "OJ" =
1619   { "journal" bibinfo.check
1620     emphasize
1621     series empty$
1622     volume empty$

```

```

1623     and
1624     pages empty$
1625     and
1626     'skip$
1627     { " " * }
1628     if$
1629     format.series
1630     volume "volume" bibinfo.check *
1631     duplicate$ empty$
1632     'skip$
1633     { pages field.or.null duplicate$ empty$
1634     'skip$
1635     { "/" swap$ *
1636     }
1637     if$ *
1638     }
1639     if$ *
1640   }
1641   { duplicate$ "ECR" =
1642     { "journal" bibinfo.check
1643     volume "volume" bibinfo.check
1644     duplicate$ empty$
1645     'skip$
1646     { " " swap$ * "--" * * }
1647     if$
1648     }
1649     { "journal" bibinfo.check

```

In bathx.bst, we conditionally include the bbl.online string:

```

1650     is.online
1651     { " " * bbl.online *
1652     #1 'online.shown :=
1653     }
1654     'skip$
1655     if$

```

```

1656     volume "volume" bibinfo.check
1657     duplicate$ empty$
1658     'pop$
1659     { " " * swap$ * }
1660     if$
1661     pages empty$
1662     eid empty$
1663     and
1664     'skip$
1665     { " " * }
1666     if$
1667     }
1668     if$
1669   }
1670   if$
1671   eid empty$
1672   { pages "pages" bibinfo.check * }
1673   { eid "eid" bibinfo.check * }
1674   if$
1675 }

```

If institution is given (not 'Commission'), format.inst.journal.series.vol.pages adds the institution, number and regular journal information to the stack. otherwise, it calls format.journal.series.vol.pages instead.

```

1676 FUNCTION {format.inst.journal.series.vol.pages}
1677 { institution duplicate$ empty$
1678   { pop$ format.journal.series.vol.pages }
1679   { duplicate$ "Commission" =
1680     { pop$ format.journal.series.vol.pages }
1681     { number duplicate$ empty$
1682       { pop$ casenumber duplicate$ empty$
1683         'skip$
1684         { "casenumber" bibinfo.check

```



```

1685         ", " swap$ * *
1686     }
1687     if$
1688 }
1689 { "number" bibinfo.check
1690     ", " swap$ * *
1691 }
1692 if$
1693 output new.sentence
1694 journal "t" change.case$
1695 "journal" bibinfo.check
1696 emphasize output

```

In `bathx.bst`, we conditionally include the `bbl.online` string:

```

1697 is.online
1698 { continue.clause
1699     bbl.online output
1700     #1 'online.shown :=
1701 }
1702 'skip$
1703 if$

```

```

1704 volume field.or.null
1705 duplicate$ empty$ 'skip$
1706 { "volume" bibinfo.check
1707 }
1708 if$
1709 eventyear field.or.null duplicate$ empty$
1710 { pop$
1711     month "month" bibinfo.check duplicate$ empty$
1712     'pop$
1713     { swap$ duplicate$ empty$
1714         'pop$
1715         { ", " * swap$ * }
1716     }
1717     if$
1718 }
1719 { swap$ duplicate$ empty$
1720     'pop$
1721     { ", " * swap$ * }
1722     if$
1723 }
1724 if$
1725 eid empty$
1726 { format.journal.pages }
1727 { format.journal.eid }
1728 if$
1729 }
1730 }
1731 if$
1732 }
1733 if$
1734 }

```

### 7.6.7 Access and URLs

`format.eprint` returns a string `\eprint[archive]{eprint}`.

```

1735 FUNCTION {format.eprint}
1736 { eprint duplicate$ empty$
1737     'skip$
1738     { "\eprint"
1739         archive empty$
1740         'skip$
1741         { "[" * archive * "]" * }
1742         if$
1743         "[" * swap$ * "]" *
1744     }
1745     if$
1746 }

```

`format.url` returns a string Available from: `\url{url}` [Accessed `urldate`]. We also allow the `library` field to be used as a fallback URL.

```

1747 FUNCTION {format.url}
1748 { doi empty$
1749   { url }
1750   { "https://doi.org/" doi * }
1751   if$
1752   duplicate$ empty$
1753     { pop$ "" }
1754     { "\urlprefix\url{" swap$ * "}" * }
1755   if$
1756   duplicate$ empty$
1757     urlyear empty$ not
1758     urldate empty$ not
1759     or
1760     and
1761     { library duplicate$ empty$
1762       'pop$
1763       { "\urlprefix{" swap$ * * }
1764       if$
1765     }
1766     'skip$
1767   if$
1768   urlyear empty$
1769     { urldate empty$
1770       'skip$
1771       { " [\urldateprefix{" * urldate * "}" * }
1772       if$
1773     }
1774     { " [\urldateprefix{" * urlyear * "}" * }
1775     if$
1776 }

```

### 7.6.8 Related items

The following functions are used for entries that look like ‘Author, (Year). Title. In: Author (Year).’ In practice, they should never be used with Harvard (Bath).

```

1777 FUNCTION {format.article.crossref}
1778 { word.in
1779   " \cite{" * crossref * "}" *
1780 }
1781 FUNCTION {format.incoll.inproc.crossref}
1782 { word.in
1783   " \cite{" * crossref * "}" *
1784 }

```

`format.book.crossref` is a bit different as it is intended for cases where the given book is volume X of a larger work: ‘Author, (Year). Title. Vol. volume of Author (Year).’ In the absence of a volume, writes a warning to the log and behaves like the above.

```

1785 FUNCTION {format.book.crossref}
1786 { volume duplicate$ empty$
1787   { "empty volume in " cite$ * "'s crossref of " * crossref * warning$
1788     pop$ word.in
1789   }
1790   { bbl.volume
1791     capitalize
1792     swap$ tie.or.space.prefix "volume" bibinfo.check * * bbl.of space.word *
1793   }
1794   if$
1795   " \cite{" * crossref * "}" *
1796 }

```

## 7.7 Drivers for formatting specific entry types

These functions are called when an entry of the given type is being formatted.

### 7.7.1 Article

```
1798 FUNCTION {article}
1799 { output.bibitem
1800   format.authors "author" output.check
1801   author format.key output
1802   format.date "year" output.check
1803   date.block
1804   format.title "title" output.check
1805   print.title.addenda
1806   new.block
1807   crossref missing$
1808     { journal "t" change.case$
1809       "journal" bibinfo.check
1810       emphasize
1811       "journal" output.check
```

In `bathx.bst`, we conditionally include the `bbl.online` string:

```
1812   is.online
1813     { continue.clause
1814       bbl.online output
1815       #1 'online.shown :=
1816     }
1817     'skip$
1818   if$
```

```
1819     format.vol.num.pages output
1820   }
1821   { format.article.crossref output.nonnull
1822     format.pages output
1823   }
1824   if$
1825   new.block
1826   format.note output
1827   new.block
1828   format.eprint output
1829   format.url output
1830   fin.entry
1831 }
```

### 7.7.2 Book

```
1832 FUNCTION {book}
1833 { output.bibitem
1834   author empty$
1835     { editor empty$
1836       type$ "reference" =
1837       or
1838         { type$ "reference" =
1839           { format.title "title" output.check }
1840           { format.btitle "title" output.check }
1841         if$
1842         print.labeltitle.addenda
1843         format.date "year" output.check
1844         year empty$ 'skip$ { date.block } if$
1845         print.title.addenda
1846       }
1847     { format.editors "author and editor" output.check
1848       format.date "year" output.check
1849       date.block
1850       format.btitle "title" output.check
1851       print.title.addenda
1852     }
1853   if$
1854 }
1855 { format.authors output.nonnull
1856   crossref missing$
1857     { "author and editor" editor either.or.check }
1858     'skip$
1859   if$
```

```

1860     format.date "year" output.check
1861     date.block
1862     format.btitle "title" output.check
1863     print.title.addenda
1864   }
1865   if$
1866   new.block
1867   format.edition output
1868   edition empty$
1869     { mid.sentence 'output.state := }
1870     'new.block
1871   if$
1872   format.bvolume output
1873   format.number.series output
1874   new.block
1875   format.note output
1876   crossref missing$
1877     { keywords field.or.null "uksi" =
1878       { mid.sentence 'output.state := }
1879       { new.sentence }
1880     if$
1881     format.publisher.address output
1882   }
1883   { new.block
1884     format.book.crossref output.nonnull
1885   }
1886   if$
1887   new.block
1888   format.eprint output
1889   format.url output
1890   fin.entry
1891 }

```

### 7.7.3 Unpublished or self-published monograph

```

1892 FUNCTION {booklet}
1893 { output.bibitem
1894   author empty$
1895   { format.avtitle "title" output.check
1896     print.labeltitle.addenda
1897     format.date "year" output.check
1898     year empty$ 'skip$ { date.block } if$
1899     print.title.addenda
1900   }
1901   { format.authors output
1902     author format.key output
1903     format.date "year" output.check
1904     date.block
1905     format.avtitle "title" output.check
1906     print.title.addenda
1907   }
1908   if$
1909   new.block
1910   format.series.episode output

```

In `bathx.bst`, we conditionally include the `bbl.online` string:

```

1911 is.online
1912   { continue.clause
1913     bbl.online output
1914     #1 'online.shown :=
1915   }
1916   'skip$
1917   if$

```

```

1918   new.block
1919   type "type" bibinfo.check output
1920   new.block
1921   howpublished "howpublished" bibinfo.check output
1922   new.block
1923   format.note output

```

```

1924 publisher empty$ 'skip$
1925   { new.block}
1926 if$
1927 format.publisher.address output
1928 new.block
1929 format.eprint output
1930 format.url output
1931 fin.entry
1932 }

```

#### 7.7.4 Work in book

```

1933 FUNCTION {inbook}
1934 { output.bibitem
1935   author empty$
1936   { format.title "title" output.check
1937     print.labeltitle.addenda
1938     format.date "year" output.check
1939     year empty$ 'skip$ { date.block } if$
1940     print.title.addenda
1941   }
1942   { format.authors output.nonnull
1943     format.date "year" output.check
1944     date.block
1945     format.title "title" output.check
1946     print.title.addenda
1947   }
1948   if$
1949   new.block
1950   crossref missing$
1951     { format.in.ed.booktitle "booktitle" output.check
1952       new.block
1953       format.edition output
1954       edition empty$
1955         { mid.sentence 'output.state := }
1956         'new.block
1957       if$
1958       format.bvolume output
1959       format.number.series output
1960       new.block
1961       format.note output
1962       new.block
1963       format.publisher.address output
1964       format.chapter.pages output
1965     }
1966     { format.note output
1967       new.block
1968       format.incoll.inproc.crossref output.nonnull
1969       format.chapter.pages output
1970     }
1971   if$
1972   new.block
1973   format.eprint output
1974   format.url output
1975   fin.entry
1976 }

```

#### 7.7.5 Work in collection

```

1977 FUNCTION {incollection}
1978 { output.bibitem
1979   author empty$
1980   { type$ "inreference" =
1981     { format.label.booktitle output}
1982     { format.title "title" output.check
1983       print.labeltitle.addenda
1984     }
1985     if$
1986     format.date "year" output.check
1987     year empty$ 'skip$ { date.block } if$
1988     type$ "inreference" =

```

```

1989     'skip$
1990     { print.title.addenda }
1991     if$
1992   }
1993   { format.authors output.nonnull
1994     format.date "year" output.check
1995     date.block
1996     format.title "title" output.check
1997     print.title.addenda
1998   }
1999   if$
2000   new.block
2001   crossref missing$
2002     { type$ "inreference" =
2003       author empty$
2004       and
2005         { format.byeditors output }
2006         { format.in.ed.booktitle "booktitle" output.check }
2007     if$
2008     new.block
2009     format.edition output
2010     type$ "inreference" =
2011     author empty$
2012     and
2013       { new.block
2014         format.btitle "title" output.check
2015         print.title.addenda
2016         new.block
2017       }
2018     { edition empty$
2019       { mid.sentence 'output.state := }
2020       'new.block
2021     if$
2022     }
2023   if$
2024   format.bvolume output
2025   format.number.series output
2026   new.block
2027   format.note output
2028   new.block
2029   format.publisher.address output
2030   format.chapter.pages output
2031   }
2032   { format.note output
2033     new.block
2034     format.incoll.inproc.crossref output.nonnull
2035     format.chapter.pages output
2036   }
2037   if$
2038   new.block
2039   format.eprint output
2040   format.url output
2041   fin.entry
2042 }

```

### 7.7.6 Conference paper

```

2043 FUNCTION {inproceedings}
2044 { output.bibitem
2045   format.authors "author" output.check
2046   author format.key output
2047   format.date "year" output.check
2048   date.block
2049   format.title "title" output.check
2050   print.title.addenda
2051   new.block
2052   crossref missing$
2053     { format.in.ed.booktitle "booktitle" output.check
2054       format.bvolume output
2055       format.number.series output
2056       eventyear output
2057       venue output

```

```

2058     format.note output
2059     new.sentence
2060     publisher empty$
2061     { format.organization.address output }
2062     { organization "organization" bibinfo.check output
2063       format.publisher.address output
2064     }
2065     if$
2066     format.pages output
2067   }
2068   { format.note output
2069     new.block
2070     format.incoll.inproc.crossref output.nonnull
2071     format.pages output
2072   }
2073   if$
2074   new.block
2075   format.eprint output
2076   format.url output
2077   fin.entry
2078 }
2079 FUNCTION {conference} { inproceedings }

```

### 7.7.7 Legal case studies

```

2080 FUNCTION {jurisdiction}
2081 { output.bibitem
2082   author empty$
2083   { format.btitle "title" output.check
2084     continue.clause
2085     format.case.number output
2086     continue.clause
2087     print.labeltitle.addenda
2088     continue.clause
2089     format.jur.date "year" output.check
2090     year empty$ 'skip$ { case.check } if$
2091     print.title.addenda
2092   }
2093   { format.authors output
2094     author format.key output
2095     format.jur.date "year" output.check
2096     date.block
2097     continue.clause
2098     format.btitle "title" output.check
2099     print.labeltitle.addenda
2100     print.title.addenda
2101   }
2102   if$
2103   case.check
2104   format.note output
2105   note empty$
2106   'case.check
2107   'new.block
2108   if$
2109   crossref missing$
2110   { continue.clause
2111     format.inst.journal.series.vol.pages output
2112   }
2113   { format.article.crossref output.nonnull
2114     format.pages output
2115   }
2116   if$
2117   new.block
2118   format.eprint output
2119   format.url output
2120   fin.entry
2121 }

```

## 7.7.8 Legislation

```
2122 FUNCTION {uklegislation}
2123 { output.bibitem
2124   author empty$
2125   { format.btitle "title" output.check
2126     print.labeltitle.addenda
2127     continue.clause
2128     format.date emphasize "year" output.check
2129     print.title.addenda
2130   }
2131   { format.authors output
2132     author format.key output
2133     format.date "year" output.check
2134     date.block
2135     continue.clause
2136     format.btitle "title" output.check
2137     print.labeltitle.addenda
2138     print.title.addenda
2139   }
2140   if$
2141   entrysubtype field.or.null "secondary" =
2142   { number "number" bibinfo.check output }
2143   { series empty$ type empty$ and
2144     { number empty$
2145       { format.chapter output }
2146       { continue.clause
2147         format.series.number.chapter output
2148       }
2149       if$
2150     }
2151     { chapter empty$
2152       { new.block }
2153       { continue.clause }
2154       if$
2155       format.series.number.chapter output
2156     }
2157     if$
2158   }
2159   if$
2160   note empty$
2161   'skip$
2162   { new.block
2163     format.note output
2164     new.block
2165   }
2166   if$
2167   publisher empty$ 'skip$
2168   { entrysubtype field.or.null "secondary" =
2169     'skip$
2170     { new.block }
2171     if$
2172     format.publisher.address output
2173   }
2174   if$
2175   new.block
2176   format.eprint output
2177   format.url output
2178   fin.entry
2179 }
2180 FUNCTION {eulegislation}
2181 { output.bibitem
2182   author empty$
2183   { format.title "title" output.check
2184     print.labeltitle.addenda
2185     continue.clause
2186     format.eu.date "year" output.check
2187     print.title.addenda
2188   }
2189   { format.authors output
2190     author format.key output
2191     format.eu.date "year" output.check
2192     date.block
```



```

2193     continue.clause
2194     format.title "title" output.check
2195     print.labeltitle.addenda
2196     print.title.addenda
2197 }
2198 if$
2199 eu.case.check
2200 format.note output
2201 note empty$
2202     'eu.case.check
2203     'new.block
2204 if$
2205 crossref missing$
2206     { continue.clause
2207       format.journal.series.vol.pages output
2208     }
2209     { format.article.crossref output.nonnull
2210       format.pages output
2211     }
2212 if$
2213 new.block
2214 format.eprint output
2215 format.url output
2216 fin.entry
2217 }
2218 FUNCTION {legislation}
2219 { journal empty$
2220     'uklegislation
2221     'eulegislation
2222 if$
2223 }

```

### 7.7.9 Parliamentary debate

It would be more efficient to use a straight alias here, but we anticipate using this driver for other templates in future. When that happens, the two sides of the test will be different.

```

2224 FUNCTION {legal}
2225 { entrysubtype field.or.null
2226     "parliamentary" =
2227     'article
2228     'article
2229 if$
2230 }

```

### 7.7.10 Manual

```

2231 FUNCTION {manual}
2232 { output.bibitem
2233     author empty$
2234     { format.btitle "title" output.check
2235       print.labeltitle.addenda
2236       format.date "year" output.check
2237       year empty$ 'skip$ { date.block } if$
2238       print.title.addenda
2239     }
2240     { format.authors output
2241       author format.key output
2242       format.date "year" output.check
2243       date.block
2244       format.btitle "title" output.check
2245       print.title.addenda
2246     }
2247 if$
2248 month "month" bibinfo.check output
2249 format.edition output
2250 new.block
2251 format.manual.number output
2252 new.block
2253 format.note format.journal.pages output
2254 new.block

```

```

2255 format.library output
2256 organization address new.block.checkb
2257 format.organization.address output
2258 new.block
2259 format.eprint output
2260 format.url output
2261 fin.entry
2262 }

```

### 7.7.11 Image

```

2263 FUNCTION {image}
2264 { output.bibitem
2265   author empty$
2266   { format.btitle "title" output.check
2267     print.labeltitle.addenda
2268     format.date "year" output.check
2269     year empty$ 'skip$ { date.block } if$
2270     print.title.addenda
2271   }
2272   { format.authors output
2273     author format.key output
2274     format.date "year" output.check
2275     date.block
2276     format.btitle "title" output.check
2277     print.title.addenda
2278   }
2279   if$
2280   format.edition output
2281   new.block
2282   format.note output
2283   new.block
2284   format.manual.number output
2285   new.block
2286   format.pub.org.lib.address output
2287   new.block
2288   format.eprint output
2289   format.url output
2290   fin.entry
2291 }

```

### 7.7.12 Master's thesis

```

2292 FUNCTION {mastersthesis}
2293 { output.bibitem
2294   format.authors "author" output.check
2295   author format.key output
2296   format.date "year" output.check
2297   date.block
2298   format.btitle "title" output.check
2299   print.title.addenda
2300   new.block
2301   format.note output
2302   new.block
2303   bbl.mthesis format.thesis.type output.nonnull
2304   new.block
2305   school "school" bibinfo.warn output
2306   address "address" bibinfo.check output
2307   new.block
2308   format.eprint output
2309   format.url output
2310   fin.entry
2311 }

```

### 7.7.13 Miscellaneous

```
2312 FUNCTION {misc}
2313 { output.bibitem
2314   author empty$
2315   { format.title "title" output.check
2316     print.labeltitle.addenda
2317     format.date "year" output.check
2318     year empty$ 'skip$ { date.block } if$
2319     print.title.addenda
2320   }
2321   { format.authors output
2322     author format.key output
2323     format.date "year" output.check
2324     date.block
2325     format.title "title" output.check
2326     print.title.addenda
2327   }
2328   if$
2329   new.block
2330   howpublished "howpublished" bibinfo.check output
2331   new.block
2332   format.note output
2333   new.block
2334   format.eprint output
2335   format.url output
2336   fin.entry
2337 }
```

### 7.7.14 Doctoral thesis

```
2338 FUNCTION {phdthesis}
2339 { output.bibitem
2340   format.authors "author" output.check
2341   author format.key output
2342   format.date "year" output.check
2343   date.block
2344   format.btitle "title" output.check
2345   print.title.addenda
2346   new.block
2347   format.note output
2348   new.block
2349   bbl.phdthesis format.thesis.type output.nonnull
2350   new.block
2351   school "school" bibinfo.warn output
2352   address "address" bibinfo.check output
2353   new.block
2354   format.eprint output
2355   format.url output
2356   fin.entry
2357 }
```

### 7.7.15 Conference proceedings

```
2358 FUNCTION {proceedings}
2359 { output.bibitem
2360   editor empty$
2361   { format.btitle "title" output.check
2362     print.labeltitle.addenda
2363     format.date "year" output.check
2364     year empty$ 'skip$ { date.block } if$
2365     print.title.addenda
2366   }
2367   { format.editors output
2368     editor format.key output
2369     format.date "year" output.check
2370     date.block
2371     format.btitle "title" output.check
2372     print.title.addenda
2373   }
2374   if$
```

```

2375 format.bvolume output
2376 format.number.series output
2377 eventyear output
2378 venue output
2379 format.note output
2380 new.sentence
2381 publisher empty$
2382 { format.organization.address output }
2383 { organization "organization" bibinfo.check output
2384   format.publisher.address output
2385 }
2386 if$
2387 new.block
2388 format.eprint output
2389 format.url output
2390 fin.entry
2391 }

```

### 7.7.16 Report

```

2392 FUNCTION {techreport}
2393 { output.bibitem
2394   author empty$
2395   { format.btitle "title" output.check
2396     print.labeltitle.addenda
2397     format.date "year" output.check
2398     year empty$ 'skip$ { date.block } if$
2399     print.title.addenda
2400   }
2401   { format.authors output
2402     author format.key output
2403     format.date "year" output.check
2404     date.block
2405     format.btitle "title" output.check
2406     print.title.addenda
2407   }
2408   if$
2409   new.block
2410   format.tr.number output
2411   new.block

```

In `bathx.bst`, the `library` field is used for the repository, so the note can go in the same position as in `biblatex-bath`:

```

2412 format.note output
2413 new.block

```

In `bath.bst`, the `note` field is recommended for the repository, therefore has to go after the publisher (unless the `library` field has been used):

```

2265 library empty$
2266 'skip$
2267 { format.note output
2268   new.block
2269 }
2270 if$

```

This bit is the same.

```

2414 publisher empty$
2415 { format.institution.address output }
2416 { institution "institution" bibinfo.check output
2417   format.publisher.address output
2418 }
2419 if$
2420 new.block

```

In `bathx.bst`, the `library` field always comes after the publisher:

```

2421 format.library output

```

In `bath.bst`, we fall back to `note` if `library` is not provided:

```
2278 library empty$
2279   { format.note output }
2280   { format.library output }
2281 if$
```

The rest is the same.

```
2422 new.block
2423 format.eprint output
2424 format.url output
2425 fin.entry
2426 }
```

### 7.7.17 Unpublished works

```
2427 FUNCTION {unpublished}
2428 { output.bibitem
2429   author empty$
2430   { booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2431     print.labeltitle.addenda
2432     format.date "year" output.check
2433     year empty$ 'skip$ { date.block } if$
2434     print.title.addenda
2435   }
2436   { format.authors output
2437     author format.key output
2438     format.date "year" output.check
2439     date.block
2440     booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2441     print.title.addenda
2442   }
2443   if$
2444   new.block
2445   format.in.ed.booktitle output
2446   new.block
2447   howpublished "howpublished" bibinfo.check output
2448   new.block
2449   format.note output
2450   new.block
2451   bbl.unpublished output
2452   new.block
2453   format.eprint output
2454   format.url output
2455   fin.entry
2456 }
```

### 7.7.18 Aliases and fallbacks

Here are a list of aliases supported by this style.

```
2458 FUNCTION {letter}
2459 { journal empty$
2460   'unpublished
2461   'article
2462   if$
2463 }
2464 FUNCTION {collection} { book }
2465 FUNCTION {reference} { book }
2466 FUNCTION {inreference} { incollection }
2467 FUNCTION {audio} { booklet }
2468 FUNCTION {movie} { booklet }
2469 FUNCTION {music} { booklet }
2470 FUNCTION {video} { booklet }
2471 FUNCTION {dataset} { manual }
2472 FUNCTION {electronic} { manual }
2473 FUNCTION {online} { manual }
2474 FUNCTION {patent} { manual }
2475 FUNCTION {software} { manual }
```

```

2476 FUNCTION {standard} { manual }
2477 FUNCTION {www} { manual }
2478 FUNCTION {thesis} { phdthesis }
2479 FUNCTION {report} { techreport }

```

The fallback driver is `misc`, as with most styles.

```

2480 FUNCTION {default.type} { misc }

```

## 7.8 Writing out the bibliography to a file

We are now ready for BibTeX to read in the `.bib` file.

```

2482 READ

```

These functions are used to generate a sort key for each entry in the `.bib` file. `sortify` strips out L<sup>A</sup>T<sub>E</sub>X commands, spaces and ASCII symbols, and converts to lowercase.

```

2483 FUNCTION {sortify}
2484 { purify$
2485   "l" change.case$
2486 }

```

`chop.word` takes a string `s`, an integer `len`, and a prefix (stop) string. If the first `len` characters of `s` match the prefix, those characters are stripped off `s`.

```

2487 INTEGERS { len }
2488 FUNCTION {chop.word}
2489 { 's :=
2490   'len :=
2491   s #1 len substring$ =
2492     { s len #1 + global.max$ substring$ }
2493     's
2494   if$
2495 }

```

`format.lab.names` is similar to `format.full.names`, the main difference being that if there are more than 3 names in the list, the second and subsequent names are replaced with ‘et al.’

```

2496 FUNCTION {format.lab.names}
2497 {'s :=
2498   "" 't :=
2499   #1 'nameptr :=
2500   s num.names$ 'numnames :=
2501   numnames 'namesleft :=
2502   { namesleft #0 > }
2503   { s nameptr
2504     "{vv~}{ll}" format.name$
2505     't :=
2506     nameptr #1 >
2507     { nameptr #2 =
2508       numnames #3 > and
2509       { "others" 't :=
2510         #1 'namesleft := }
2511       'skip$
2512       if$
2513       namesleft #1 >
2514       { ", " * t * }
2515       { s nameptr "{ll}" format.name$ duplicate$ "others" =
2516         { 't := }
2517         { pop$ }
2518         if$
2519         t "others" =
2520         { " " * bbl.etal *
2521           }
2522         { bbl.and
2523           space.word * t *
2524           }
2525         if$

```

```

2526     }
2527     if$
2528   }
2529   't
2530   if$
2531   nameptr #1 + 'nameptr :=
2532   namesleft #1 - 'namesleft :=
2533   }
2534   while$
2535 }

```

author.editor.key.label is the analogue of author.editor.key.full but with a truncated list of author/editor names.

```

2536 FUNCTION {author.editor.key.label}
2537 { author empty$
2538   { editor empty$
2539     { key empty$
2540       { title empty$
2541         { cite$ #1 #3 substring$ }
2542         { title
2543           type$ "book" =
2544             { emphasize }
2545             'skip$
2546           if$
2547         }
2548         if$
2549       }
2550       'key
2551     if$
2552   }
2553   { editor format.lab.names }
2554   if$
2555 }
2556 { author format.lab.names }
2557 if$
2558 }

```

editor.key.label is the analogue of editor.key.full but with a truncated list of editor names.

```

2559 FUNCTION {editor.key.label}
2560 { editor empty$
2561   { key empty$
2562     { title empty$
2563       { cite$ #1 #3 substring$ }
2564       { title emphasize }
2565     if$
2566   }
2567   'key
2568   if$
2569 }
2570 { editor format.lab.names }
2571 if$
2572 }

```

author.key.label is the analogue of author.key.full but with a truncated list of author names.

```

2573 FUNCTION {author.key.label}
2574 { author empty$
2575   { key empty$
2576     { type$ "inreference" =
2577       booktitle empty$ not
2578       and
2579       { booktitle }
2580       { title empty$
2581         { cite$ #1 #3 substring$ }
2582         { title
2583           type$ "article" =
2584           type$ "incollection" =

```

```

2585         or
2586         type$ "inproceedings" =
2587         or
2588         type$ "reference" =
2589         author empty$
2590         and
2591         or
2592         type$ "misc" =
2593         or
2594         'skip$
2595         { emphasize }
2596         if$
2597     }
2598     if$
2599 }
2600     if$
2601 }
2602     'key
2603     if$
2604 }
2605 { author format.lab.names }
2606 if$
2607 }

```

calc.short.authors chooses between the above.

```

2608 FUNCTION {calc.short.authors}
2609 { type$ "book" =
2610   type$ "inbook" =
2611   or
2612     'author.editor.key.label
2613     { type$ "proceedings" =
2614       'editor.key.label
2615       'author.key.label
2616       if$
2617     }
2618   if$
2619   'short.list :=
2620 }

```

calc.label creates a string in the form ‘Truncated Author List(Year’ and saves it to the label variable. The closing parenthesis is supplied by output.bibitem (see section 7.5), into which this string is incorporated.

```

2621 FUNCTION {calc.label}
2622 { calc.short.authors
2623   short.list
2624   "("
2625   *
2626   year nodate.check duplicate$ empty$
2627   { pop$ "" }
2628   'skip$
2629   if$
2630   *
2631   'label :=
2632 }

```

sort.format.names consumes one token (a field) and constructs a list of names in the form ‘Surname Forename Prefix Suffix’, with just spaces between adjacent names and ‘zzzzz’ used in place of ‘et al.’

```

2633 FUNCTION {sort.format.names}
2634 { 's :=
2635   #1 'nameptr :=
2636   ""
2637   s num.names$ 'numnames :=
2638   numnames 'namesleft :=
2639   { namesleft #0 > }
2640   { s nameptr
2641     "{ll{ }}{ f{ }}{ jj{ }}{ vv{ }}"
2642     format.name$ 't :=

```



```

2643     nameptr #1 >
2644     { " " *
2645       namesleft #1 =
2646       t "others" =
2647       and
2648         { "zzzz" 't := }
2649         'skip$
2650         if$
2651         t sortify *
2652       }
2653     { t sortify * }
2654     if$
2655     nameptr #1 + 'nameptr :=
2656     namesleft #1 - 'namesleft :=
2657   }
2658   while$
2659 }

```

sort.format.title strips off initial articles from the preceding field and normalizes what remains using sortify (see above).

```

2660 FUNCTION {sort.format.title}
2661 { 't :=
2662   "A " #2
2663   "An " #3
2664   "The " #4 t chop.word
2665   chop.word
2666   chop.word
2667   sortify
2668   #1 global.max$ substring$
2669 }

```

author.editor.sort creates a string for use when sorting the entry in the bibliography. In order of preference, the string is based on author, editor, title or the key field.

```

2670 FUNCTION {author.editor.sort}
2671 { author empty$
2672   { editor empty$
2673     { title empty$
2674       { key empty$
2675         { "to sort, need author, editor, title, or key in " cite$ * warning$
2676           ""
2677         }
2678         { key sortify }
2679       } if$
2680     }
2681     { title sort.format.title }
2682   } if$
2683 }
2684 { editor sort.format.names }
2685 } if$
2686 }
2687 { author sort.format.names }
2688 } if$
2689 }

```

editor.sort does similar but skips the author names.

```

2690 FUNCTION {editor.sort}
2691 { editor empty$
2692   { title empty$
2693     { key empty$
2694       { "to sort, need editor, title, or key in " cite$ * warning$
2695         ""
2696       }
2697       { key sortify }
2698     } if$
2699   }
2700   { title sort.format.title }
2701 } if$
2702 }

```

```

2703     { editor sort.format.names }
2704   if$
2705 }

```

author.sort does similar but skips the editor names.

```

2706 FUNCTION {author.sort}
2707 { author empty$
2708   { type$ "inreference" =
2709     booktitle empty$ not
2710     and
2711     { booktitle sort.format.title }
2712     { title empty$
2713       { key empty$
2714         { "to sort, need author, title, or key in " cite$ * warning$
2715           ""
2716         }
2717         { key sortify }
2718         if$
2719       }
2720     { title sort.format.title }
2721     if$
2722   }
2723   if$
2724 }
2725 { author sort.format.names }
2726 if$
2727 }

```

presort constructs the sort.label from the calc.label and one of the above three functions. It also constructs the sort.key\$ (used by SORT) from the sort.label and the title.

```

2729 FUNCTION {presort}
2730 { calc.label
2731   label sortify
2732   " "
2733   *
2734   type$ "book" =
2735   type$ "inbook" =
2736   or
2737   'author.editor.sort
2738   { type$ "proceedings" =
2739     'editor.sort
2740     'author.sort
2741     if$
2742   }
2743   if$
2744   #1 entry.max$ substring$
2745   'sort.label :=
2746   sort.label
2747   *
2748   " "
2749   *
2750   title field.or.null
2751   sort.format.title
2752   *
2753   #1 entry.max$ substring$
2754   'sort.key$ :=
2755 }

```

We now tell BibTeX to go through each entry in turn and apply the presort function, then use the resultant sort keys to put the entries in order.

```

2757 ITERATE {presort}
2758 SORT

```

If there are any duplicate author–year combinations, these functions detect them and disambiguate them by adding a letter to the year.

```

2759 STRINGS { last.label next.extra }
2760 INTEGERS { last.extra.num last.extra.num.extended last.extra.num.blank number.label }
2761 FUNCTION {initialize.extra.label.stuff}
2762 { #0 int.to.chr$ 'last.label :=
2763   "" 'next.extra :=
2764   #0 'last.extra.num :=
2765   "a" chr.to.int$ #1 - 'last.extra.num.blank :=
2766   last.extra.num.blank 'last.extra.num.extended :=
2767   #0 'number.label :=
2768 }
2769 FUNCTION {forward.pass}
2770 { last.label label =
2771   { last.extra.num #1 + 'last.extra.num :=
2772     last.extra.num "z" chr.to.int$ >
2773     { "a" chr.to.int$ 'last.extra.num :=
2774       last.extra.num.extended #1 + 'last.extra.num.extended :=
2775     }
2776     'skip$
2777     if$
2778     last.extra.num.extended last.extra.num.blank >
2779     { last.extra.num.extended int.to.chr$
2780       last.extra.num int.to.chr$
2781       * 'extra.label := }
2782     { last.extra.num int.to.chr$ 'extra.label := }
2783     if$
2784   }
2785   { "a" chr.to.int$ 'last.extra.num :=
2786     "" 'extra.label :=
2787     label 'last.label :=
2788   }
2789   if$
2790   number.label #1 + 'number.label :=
2791 }
2792 FUNCTION {reverse.pass}
2793 { next.extra "b" =
2794   { "a" 'extra.label := }
2795   'skip$
2796   if$
2797   extra.label 'next.extra :=
2798   extra.label
2799   duplicate$ empty$
2800   'skip$
2801   { "{\natexlab{" swap$ * "}" * }
2802   if$
2803   'extra.label :=
2804   label extra.label * 'label :=
2805 }

```

Applying these functions is a bit more tricky, but once we have done that, we re-sort the entries to make absolutely sure they in the right order, this time with the year between with `sort.label` and the title.

```

2806 EXECUTE {initialize.extra.label.stuff}
2807 ITERATE {forward.pass}
2808 REVERSE {reverse.pass}
2809 FUNCTION {bib.sort.order}
2810 { sort.label
2811   " "
2812   *
2813   year field.or.null sortify
2814   *
2815   " "
2816   *
2817   title field.or.null
2818   sort.format.title
2819   *
2820   #1 entry.max$ substring$
2821   'sort.key$ :=
2822 }
2823 ITERATE {bib.sort.order}
2824 SORT

```

We now write out the beginning of the bibliography to a .bbl file. This includes opening a L<sup>A</sup>T<sub>E</sub>X environment, `thebibliography`, and some user-redefinable strings.

```

2825 FUNCTION {begin.bib}
2826 { preamble$ empty$
2827   'skip$
2828   { preamble$ write$ newline$ }
2829   if$
2830   "\begin{thebibliography}{ " number.label int.to.str$ * " }" *
2831   write$ newline$
2832   "\providecommand{\natexlab}[1]{#1}"
2833   write$ newline$
2834   "\providecommand{\url}[1]{\texttt{#1}}"
2835   write$ newline$
2836   "\providecommand{\urlprefix}{Available from: }"
2837   write$ newline$
2838   "\providecommand{\urldateprefix}{Accessed }"
2839   write$ newline$
2840   "\providecommand{\selectlanguage}[1]{\relax}"
2841   write$ newline$
2842   "\providecommand{\bibinfo}[2]{#2}"
2843   write$ newline$
2844   "\providecommand{\eprint}[2][ ]{\url{#2}}"
2845   write$ newline$
2846 }
2847 EXECUTE {begin.bib}
2848 EXECUTE {init.state.consts}

```

We go through each entry in turn, in the right order, and write the formatted text out to the file.

```

2849 ITERATE {call.type$}

```

After the last entry, we close the L<sup>A</sup>T<sub>E</sub>X environment and stop writing to the .bbl file.

```

2850 FUNCTION {end.bib}
2851 { newline$
2852   "\end{thebibliography}" write$ newline$
2853 }
2854 EXECUTE {end.bib}

```