

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

Maintainer: Alex Ball*

Package v6.0 – 25 May 2022

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.etal2021ipp},
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_TEX. 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.

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

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

RX

British National Formulary, 2019. *Aspirin* [Online]. London: Pharmaceutical Press. Available from: 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 [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 Bib_TEX and biblatex, one unfortunate difference is that you must escape the hash symbol inside URLs with Bib_TEX 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. *Brit. Med. J.*, 341(7765), pp.157–208.
- ⚙️ Newman, R., 2010. Malaria control beyond 2010. *Brit. Med. J.*, 341(7765), pp.157–208.

```
@article{newman2010mcb,
  author = {Newman, R.},
  year = {2010},
  title = {Malaria control beyond 2010},
  journal = {{Brit.\@ Med.\@ J.}},
  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 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].
- ⚙️ 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].

```
@article{devlin.etal2021ipp,
  author = {Devlin, S. M. and Martin, A. and Ostrovnya, 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 `biblatex`) 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_T_EX, but are needed if you want compatibility with bib_l_a_t_e_x.

- 📖 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 BibTeX 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 [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 [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://www.youtube.com/watch?v=gKTfCz4JtVE&feature=youtu.be> [Accessed 29 April 2016].
- ⚙️ Moran, C., 2016. *Save our libraries* [Online]. Available from: <https://www.youtube.com/watch?v=gKTfCz4JtVE&feature=youtu.be> [Accessed 29 April 2016].

```
@video{moran2016sol,  
  author = {Moran, C.},  
  year = {2016},  
  title = {Save Our Libraries},  
  url = {https://www.youtube.com/watch?v=gKTfCz4JtVE&feature=youtu.be},  
  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


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 [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 [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 [Accessed 30 July 2004].

⚙️ Clark, T., 5 July 2004. A European UK Libraries Plus? *Lis-link* [Online]. Available from: 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 Legal or government document: UK parliamentary reports/legislation

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.

```
@techreport{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},
  type = {},
  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.

```
@techreport{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},  
  type = {},  
  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.

```
@techreport{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},  
  type = {},  
  number = {6041}}
```

RX

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 [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 [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.

5.9 Legal or government document: EU reports/legislation

EU publication

- 📄 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 or government document: case reports

Legal case study

- 📄 *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, 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 `bibtex` 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 = {2012},
note = {[2012]. UKSC 16}
```

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 (`bibtex-oscola`) `userb` field for the Commission case number, `number` for the formal decision number, and give 'Commission' as the `institution`.

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].
- 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 [Accessed 26 November 2019].
- British National Formulary, 2020a. 79th ed. *Aspirin*. London: Pharmaceutical Press.
- 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].
- Clark, T., 5 July 2004. A European UK Libraries Plus? *Lis-link* [Online]. Available from: 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) 1831/2003 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 Ostrovskaya, 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 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/ukxi/2012/1916/pdfs/ukxi_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 [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://www.youtube.com/watch?v=gKTfCz4JtVE&feature=youtu.be> [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. *Brit. Med. J.*, 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*. [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 [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 2016-2022 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 `bathtx.bst`.

7.1 Data model

We begin by listing the entry fields that we will recognize. These are the standard Bib_TE_X 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       }
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             { bbl.and
442               space.word * t * }
443             }
444           if$
445         }
446       if$
447     }
448     't
449     if$
450     nameptr #1 + 'nameptr :=
451     namesleft #1 - 'namesleft :=
452   }
453   while$
454 }
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             { bbl.and
483               space.word * t * }
484             }
485           if$
486         }
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 bbl.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.cause
773 }
774 'skip$
775 if$
776 format.version output
777 is.online
778 { entrysubtype duplicate$ empty$
779   type$ duplicate$ "legislation" =
780   swap$ "jurisdiction" =
781   or
782 or
783 { pop$
784   journal empty$
785   booktitle empty$
786   type$ "inreference" =
787   author empty$
788   and
789   or
790 and
791   library empty$
792   type$ duplicate$ "image" =
793   swap$ duplicate$ "audio" =
794   swap$ duplicate$ "video" =
795   swap$ duplicate$ "music" =
796   swap$ "movie" =
797   or
798   or
799   or
800   or
801   or
802 and
803   series empty$
804   type field.or.null
805   duplicate$ "TV" =
806   swap$ "Radio" =
807   or not
808   or
809 and
810   { continue.cause
811     bbl.online output
812     #1 'online.shown :=
813   }
814   'skip$
815 if$
816 }
817 { "entrysubtype" bibinfo.check
818   "[" swap$ * "]" *
819   continue.cause output
820 }
821 if$
822 }
823 'skip$
824 if$

```

We finish off with editor/translator information.

```

825 continue.cause
826 type$ "reference" =
827 { format.byeditors output }
828 'skip$
829 if$
830 format.translators output
831 }

```

format.label.booktitle returns the booktitle field upright

```

832 FUNCTION {format.label.booktitle}
833 { booktitle
834   duplicate$ empty$ 'skip$

```



```

835 { "t" change.case$ }
836 if$
837 "booktitle" bibinfo.check
838 }

```

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

```

839 FUNCTION {format.booktitle}
840 { format.label.booktitle
841   emphasize
842 }

```

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

```

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

```

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

```

848 is.online
849 { " " *
850   bbl.online *
851   #1 'online.shown :=
852 }
853 'skip$
854 if$

```

```

855 editor "editor" format.names.ed duplicate$ empty$ 'pop$
856 { ", " *
857   " " *
858   get.bbl.editor add.period$
859   " " *
860   * swap$
861   *
862   word.in swap$ *
863 }
864 if$
865 }
866 if$
867 }

```

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

```

868 FUNCTION {legal.journal.shape}
869 { duplicate$ "OJ" =
870   'emphasize
871   'skip$
872   if$
873 }

```

7.6.2 Dates

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

```

773 FUNCTION {nodate.check}
774 {
775   skip$
776 }

```

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.

```

874 FUNCTION {nodate.check}
875 {
876   duplicate$ empty$
877   sortyear empty$
878   and
879   { options field.or.null "nonodate" =

```

```

880     'skip$
881     { pop$ bbl.nodate }
882     if$
883   }
884   'skip$
885   if$
886 }

```

format.date returns the year and any disambiguation label.

```

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

```

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.

```

891 FUNCTION {format.date.brackets}
892 { format.date
893   "[" swap$ * "]" *
894 }
895 FUNCTION {format.date.parens}
896 { format.date
897   "(" swap$ * ")" *
898 }
899 FUNCTION {format.jur.date.essential}
900 { options field.or.null "scottish-style" =
901   keywords field.or.null "sc" =
902   or
903   'format.date
904   'format.date.brackets
905   if$
906 }
907 FUNCTION {format.jur.date}
908 { journal field.or.null 's :=
909   s "OJ" =
910   s "ECR" =
911   or
912   'format.date.brackets
913   { options field.or.null "year-essential=false" =
914     'format.date.parens
915     { volume empty$
916       'format.jur.date.essential
917       { options field.or.null "year-essential=true" =
918         'format.jur.date.essential
919         'format.date.parens
920         if$
921       }
922     }
923   }
924   if$
925 }
926 if$
927 }

```

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

```

928 FUNCTION {format.eu.date}
929 { journal field.or.null 's :=
930   s "OJ" =
931   s "ECR" =
932   or
933   'format.date.brackets
934   'format.date
935   if$
936 }

```

7.6.3 Series, types and numbering

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

```
937 FUNCTION {format.series}
938 { series
939   duplicate$ empty$ 'skip$
940   { "t" change.case$ }
941   if$
942   "series" bibinfo.check
943 }
```

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

```
944 FUNCTION {format.bvolume}
945 { volume empty$
946   { "" }
947   { bbl.volume volume tie.or.space.prefix
948     "volume" bibinfo.check * *
949     format.series
950     duplicate$ empty$ 'pop$
951     { emphasize ", " * swap$ * }
952     if$
953     "volume and number" number either.or.check
954   }
955   if$
956 }
```

`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.

```
957 FUNCTION {format.number.series}
958 { volume empty$
959   { number empty$
960     { series field.or.null }
961     { series empty$
962       { number "number" bibinfo.check }
963       { format.series
964         number tie.or.space.prefix "number" bibinfo.check * *
965       }
966       if$
967     }
968     if$
969   }
970   { "" }
971   if$
972 }
```

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

```
973 FUNCTION {format.series.episode}
974 { series
975   duplicate$ empty$
976   'skip$
977   { "t" change.case$ }
978   if$
979   "series" bibinfo.check emphasize
980   number "number" bibinfo.check emphasize
981   duplicate$ empty$
982   { * }
983   { swap$
984     duplicate$ empty$
985     { * }
986     { ", " * swap$ * }
987     if$
988   }
989   if$
990 }
```

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

```

991 FUNCTION {is.num}
992 { chr.to.int$
993   duplicate$ "0" chr.to.int$ < not
994   swap$ "9" chr.to.int$ > not and
995 }

```

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

```

996 FUNCTION {extract.num}
997 { duplicate$ 't :=
998   "" 's :=
999   { t empty$ not }
1000   { t #1 #1 substring$
1001     t #2 global.max$ substring$ 't :=
1002     duplicate$ is.num
1003     { s swap$ * 's := }
1004     { pop$ "" 't := }
1005     if$
1006   }
1007   while$
1008   s empty$
1009   'skip$
1010   { pop$ s }
1011   if$
1012 }

```

`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.

```

1013 FUNCTION {convert.edition}
1014 { extract.num "1" change.case$ 's :=
1015   s "first" = s "1" = or
1016   { bbl.first 't := }
1017   { s "second" = s "2" = or
1018     { bbl.second 't := }
1019     { s "third" = s "3" = or
1020       { bbl.third 't := }
1021       { s "fourth" = s "4" = or
1022         { bbl.fourth 't := }
1023         { s "fifth" = s "5" = or
1024           { bbl.fifth 't := }
1025           { s #1 #1 substring$ is.num
1026             { s eng.ord 't := }
1027             { edition 't := }
1028             if$
1029           }
1030         }
1031       }
1032     }
1033   }
1034   if$
1035 }
1036 if$
1037 }
1038 if$
1039 t
1040 }

```

`format.edition` returns the converted edition field and 'ed.'

```

1041 FUNCTION {format.edition}
1042 { edition duplicate$ empty$ 'skip$
1043   { convert.edition
1044     output.state mid.sentence =
1045     { "1" }
1046     { "t" }
1047     if$ change.case$
1048     "edition" bibinfo.check
1049     " " * bbl.edition *
1050   }
1051   if$

```

```
1052 }
```

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.

```
1053 FUNCTION {format.thesis.type}
1054 { type duplicate$ empty$
1055   'pop$
1056   { swap$ pop$
1057     "t" change.case$ "type" bibinfo.check
1058   }
1059   if$
1060 }
```

`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□' (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.

```
1061 FUNCTION {add.cmd.number}
1062 { " " swap$ "number" bibinfo.check *
1063 }
1064 FUNCTION {format.legr.number}
1065 { series field.or.null 's :=
1066   s "C" =
1067     'add.cmd.number
1068     { s "Cd" =
1069       'add.cmd.number
1070       { s "Cmd" =
1071         'add.cmd.number
1072         { s "Cmnd" =
1073           'add.cmd.number
1074           { s "Cm" =
1075             'add.cmd.number
1076             { s purify$ #1 #3 substring$ 't :=
1077               t "HL " =
1078                 { ", (" swap$
1079                   "number" bibinfo.check *
1080                   ")" *
1081                 }
1082                 { ", " swap$
1083                   "number" bibinfo.check *
1084                 }
1085               if$
1086             }
1087           if$
1088         }
1089       if$
1090     }
1091   if$
1092 }
1093 if$
1094 }
1095 if$
1096 }
1097 FUNCTION {format.tr.number}
1098 { series duplicate$ empty$
1099   { pop$
1100     type duplicate$ empty$
1101     { pop$
1102       number duplicate$ empty$
1103       { pop$ ""
1104       }
1105       { "(" swap$
1106         "number" bibinfo.check
1107         * ")" *
1108       }
1109     }
1110   }
1111 }
```

```

1109     if$
1110   }
1111   { "(" swap$
1112     "t" change.case$ "type" bibinfo.check
1113     *
1114     number duplicate$ empty$
1115     'pop$
1116     { tie.or.space.prefix
1117       "number" bibinfo.check
1118       * *
1119     }
1120     if$
1121     ")" *
1122   }
1123   if$
1124 }
1125 { "t" change.case$ "(" swap$ *
1126   type duplicate$ empty$
1127   { pop$
1128     number duplicate$ empty$
1129     'pop$
1130     { format.legr.number * }
1131     if$
1132   }
1133   { "t" change.case$ "type" bibinfo.check
1134     *
1135     number duplicate$ empty$
1136     'pop$
1137     { tie.or.space.prefix
1138       "number" bibinfo.check
1139       * *
1140     }
1141     if$
1142   }
1143   if$
1144   ")" *
1145   continue.clause
1146 }
1147 if$
1148 }

```

`format.manual.number` returns the series, type and number fields as one of 'number', 'type number', 'series number', or 'series number, type'.

```

1149 FUNCTION {format.manual.number}
1150 { series duplicate$ empty$
1151   { pop$
1152     type duplicate$ empty$
1153     { pop$
1154       number duplicate$ empty$
1155       { pop$ ""
1156       }
1157       { "number" bibinfo.check
1158       }
1159     }
1160   }
1161   { "t" change.case$ "type" bibinfo.check
1162     number duplicate$ empty$
1163     'pop$
1164     { tie.or.space.prefix
1165       "number" bibinfo.check
1166       * *
1167     }
1168     if$
1169   }
1170   if$
1171 }
1172 { "t" change.case$
1173   number duplicate$ empty$
1174   { pop$
1175   }
1176   { tie.or.space.prefix
1177     "number" bibinfo.check

```

```

1178     * *
1179     }
1180     if$
1181     type duplicate$ empty$
1182     'pop$
1183     { ", " swap$ *
1184     "t" change.case$ "type" bibinfo.check
1185     *
1186     }
1187     if$
1188     }
1189     if$
1190 }

```

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

```

1191 FUNCTION {format.chapter}
1192 { chapter duplicate$ empty$
1193   { pop$ "" }
1194   { "chapter" bibinfo.check
1195     "c." swap$ *
1196   }
1197   if$
1198 }

```

`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.

```

1199 FUNCTION {format.series.number.chapter}
1200 { series duplicate$ empty$
1201   { pop$ "(" }
1202   { "series" bibinfo.check
1203     "(" swap$ *
1204   }
1205   if$
1206   type duplicate$ empty$
1207   { pop$
1208     number duplicate$ empty$
1209     'pop$
1210     { "number" bibinfo.check
1211       swap$ duplicate$ "(" =
1212       { swap$ * }
1213       { ", " * swap$ * }
1214     }
1215     if$
1216   }
1217   chapter duplicate$ empty$
1218   'pop$
1219   { "chapter" bibinfo.check
1220     swap$ duplicate$ "(" =
1221     { swap$ * }
1222     { ", c." * swap$ * }
1223     if$
1224   }
1225   if$
1226 }
1227 { "type" bibinfo.check
1228   swap$ duplicate$ "(" =
1229   { swap$ * }
1230   { ", " * swap$ * }
1231   if$
1232   number duplicate$ empty$
1233   { pop$
1234     chapter duplicate$ empty$
1235     'pop$
1236     { "chapter" bibinfo.check
1237       " c." swap$ * *
1238     }
1239     if$
1240   }
1241   { "number" bibinfo.check
1242     " " swap$ * *
1243     chapter duplicate$ empty$

```

```

1244         'pop$
1245         { "chapter" bibinfo.check
1246           ", c." swap$ * *
1247         }
1248         if$
1249       }
1250     if$
1251   }
1252   if$
1253   ")" *
1254   duplicate$ "(" =
1255     { pop$ "" }
1256     'skip$
1257   if$
1258 }

```

`format.case.number` returns the number field, suppressing punctuation, or inserts a period. In the case of EU Commission Decisions, the format is '(userb) Commission Decision number'

```

1259 FUNCTION {format.case.number}
1260 { casenumber duplicate$ empty$
1261   { pop$
1262     number duplicate$ empty$
1263     { pop$ "" }
1264     { institution field.or.null "Commission" =
1265       userb empty$ not
1266       and
1267         { userb "userb" bibinfo.check
1268           "(" swap$ * ")" * bbl.commission * " " *
1269           swap$ "number" bibinfo.check *
1270         }
1271         { "number" bibinfo.check
1272           "(" swap$ * ")" *
1273         }
1274       if$
1275     }
1276     if$
1277   }
1278   { "casenumber" bibinfo.check
1279     "(" swap$ * ")" *
1280     institution field.or.null "Commission" =
1281     number empty$ not
1282     and
1283     { " " * bbl.commission * " " *
1284       number "number" bibinfo.check *
1285     }
1286     'skip$
1287   if$
1288 }
1289 if$
1290 }

```

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

```

1291 FUNCTION {eu.case.check}
1292 { journal field.or.null "OJ" =
1293   number empty$ not
1294   or
1295   casenumber empty$ not
1296   or
1297   'continue.clause
1298   'new.block
1299   if$
1300 }

```

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).


```

1301 FUNCTION {format.note}
1302 { note empty$
1303   { "" }
1304   { note #1 #1 substring$
1305     duplicate$ "{" =
1306       'skip$
1307       { output.state mid.sentence =
1308         output.state mid.clause =
1309         or
1310         { "l" }
1311         { "u" }
1312         if$
1313         change.case$
1314       }
1315     if$
1316     note #2 global.max$ substring$ * "note" bibinfo.check
1317   }
1318   if$
1319 }

```

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.

```

1320 FUNCTION {prepend.address}
1321 { 't :=
1322   ""
1323   address empty$ t empty$ and
1324     'skip$
1325     { address "address" bibinfo.check *
1326       t empty$
1327         'skip$
1328         { address empty$
1329           'skip$
1330           { ": " * }
1331           if$
1332           t *
1333         }
1334       if$
1335     }
1336     if$
1337 }

```

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

```

1338 FUNCTION {format.publisher.address}
1339 { publisher "publisher" bibinfo.warn prepend.address
1340 }
1341 FUNCTION {format.organization.address}
1342 { publisher empty$
1343   { organization "organization" bibinfo.check prepend.address }
1344   { publisher "publisher" bibinfo.warn prepend.address
1345     organization duplicate$ empty$
1346     'pop$
1347     { "organization" bibinfo.check ". " * swap$ * }
1348     if$
1349   }
1350   if$
1351 }
1352 FUNCTION {format.institution.address}
1353 { publisher empty$
1354   { institution "institution" bibinfo.check prepend.address }
1355   { publisher "publisher" bibinfo.warn prepend.address
1356     institution duplicate$ empty$
1357     'pop$
1358     { "institution" bibinfo.check ". " * swap$ * }
1359     if$
1360   }

```

```
1361 if$
1362 }
```

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

```
1363 FUNCTION {format.library}
1364 { library "library" bibinfo.check
1365   emphasize
```

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

```
1366 duplicate$ empty$ not
1367 is.online
1368 and
1369 journal empty$
1370 and
1371 { " " * bbl.online *
1372   #1 'online.shown :=
1373 }
1374 'skip$
1375 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`.

```
1376 }
1377 FUNCTION {format.img.library}
1378 { library "library" bibinfo.check
1379 }
1380 FUNCTION {format.img.library.address}
1381 { library duplicate$ empty$
1382   { pop$
1383     institution duplicate$ empty$
1384     'skip$
1385     { "institution" bibinfo.check }
1386     if$
1387   }
1388   { "library" bibinfo.check }
1389   if$
1390   duplicate$ empty$
1391   { address "address" bibinfo.check * }
1392   { address duplicate$ empty$
1393     'pop$
1394     { "address" bibinfo.check
1395       swap$ ", " * swap$ *
1396     }
1397     if$
1398   }
1399   if$
1400 }
1401 FUNCTION {format.pub.org.lib.address}
1402 { publisher duplicate$ empty$
1403   { pop$
1404     organization duplicate$ empty$
1405     { pop$
1406       format.img.library.address }
1407     { "organization" bibinfo.check
1408       library empty$
1409       { prepend.address }
1410       { ". " *
1411         format.img.library.address *
1412       }
1413       if$
1414     }
1415     if$
1416   }
1417   { "publisher" bibinfo.warn prepend.address
1418     library empty$
1419     'skip$
1420     { ". " *
```

```

1421     format.img.library *
1422   }
1423   if$
1424 }
1425 if$
1426 }

```

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.

```

1428 INTEGERS { multiresult }
1429 FUNCTION {multi.page.check}
1430 { 't :=
1431   #0 'multiresult :=
1432   { multiresult not
1433     t empty$ not
1434     and
1435   }
1436   { t #1 #1 substring$
1437     duplicate$ "-" =
1438     swap$ duplicate$ "," =
1439     swap$ "+" =
1440     or or
1441     { #1 'multiresult := }
1442     { t #2 global.max$ substring$ 't := }
1443     if$
1444   }
1445 while$
1446 multiresult
1447 }

```

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

```

1448 FUNCTION {n.dashify}
1449 { 't :=
1450   ""
1451   { t empty$ not }
1452   { t #1 #1 substring$ "-" =
1453     { t #2 #2 substring$ "--" = not
1454       { "--" *
1455         t #2 global.max$ substring$ 't :=
1456       }
1457       { { t #1 #1 substring$ "-" = }
1458         { "-" *
1459           t #2 global.max$ substring$ 't :=
1460         }
1461         while$
1462       }
1463     }
1464   }
1465   { t #1 #1 substring$ *
1466     t #2 global.max$ substring$ 't :=
1467   }
1468   if$
1469 }
1470 while$
1471 }

```

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

```

1472 FUNCTION {format.pages}
1473 { pages duplicate$ empty$ 'skip$
1474   { duplicate$ multi.page.check
1475     { bbl.pages swap$
1476       n.dashify
1477     }
1478     { bbl.page swap$
1479     }
1480     if$
1481     "pages" bibinfo.check

```

```

1482     *
1483   }
1484   if$
1485 }

```

`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.

```

1486 FUNCTION {format.journal.pages}
1487 { pages duplicate$ empty$ 'pop$
1488   { swap$ duplicate$ empty$
1489     { pop$ pop$ format.pages }
1490     { ", " *
1491       swap$
1492       n.dashify
1493       pages multi.page.check
1494       'bbl.pages
1495       'bbl.page
1496       if$
1497       swap$
1498       "pages" bibinfo.check
1499       * *
1500     }
1501     if$
1502   }
1503   if$
1504 }

```

`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).

```

1505 FUNCTION {format.journal.eid}
1506 { eid "eid" bibinfo.check
1507   duplicate$ empty$ 'pop$
1508   { swap$ duplicate$ empty$ 'skip$
1509     { ", " *
1510     }
1511     if$
1512     swap$ *
1513   }
1514   if$
1515 }

```

`format.vol.num.pages` returns the journal pinpointing in the form ‘volume(number), eid/pages’. 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! Since the Harvard (Bath) style places publication state information in the volume position, the `pubstate` field from `biblatex` is emulated here.

```

1516 FUNCTION {format.vol.num.pages}
1517 { pubstate field.or.null
1518   duplicate$ "inpress" =
1519   { pop$ bbl.inpress }
1520   { duplicate$ "submitted" =
1521     { pop$ bbl.submitted }
1522     { duplicate$ "inpreparation" =
1523       { pop$ bbl.inpreparation }
1524       { pop$ "" }
1525     }
1526   }
1527   if$
1528 }
1529 if$
1530 duplicate$ empty$
1531 { pop$
1532   volume field.or.null
1533   duplicate$ empty$ 'skip$
1534   { "volume" bibinfo.check
1535   }

```

```

1536     if$
1537     number "number" bibinfo.check duplicate$ empty$ 'skip$
1538         { swap$ duplicate$ empty$
1539           { "there's a number but no volume in " cite$ * warning$ }
1540           'skip$
1541           if$
1542           swap$
1543           "(" swap$ * ")" *
1544         }
1545     if$ *
1546     month "month" bibinfo.check duplicate$ empty$
1547         'pop$
1548         { swap$ duplicate$ empty$
1549           'pop$
1550           { ", " * swap$ * }
1551           if$
1552         }
1553     if$
1554     eid empty$
1555         { format.journal.pages }
1556         { format.journal.eid }
1557     if$
1558   }
1559   'skip$
1560   if$
1561 }

```

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

```

1562 FUNCTION {format.chapter.pages}
1563 { chapter empty$
1564   'format.pages
1565   { type empty$
1566     { bbl.chapter }
1567     { type "1" change.case$
1568       "type" bibinfo.check
1569     }
1570     if$
1571     chapter tie.or.space.prefix
1572     "chapter" bibinfo.check
1573     * *
1574     pages empty$
1575     'skip$
1576     { ", " * format.pages * }
1577     if$
1578   }
1579   if$
1580 }

```

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.

```

1581 FUNCTION {format.journal.series.vol.pages}
1582 { journal field.or.null
1583   duplicate$ "OJ" =
1584   { "journal" bibinfo.check
1585     emphasize
1586     series empty$
1587     volume empty$
1588     and
1589     pages empty$
1590     and
1591     'skip$
1592     { " " * }
1593     if$
1594     format.series
1595     volume "volume" bibinfo.check *
1596     duplicate$ empty$
1597     'skip$
1598     { pages field.or.null duplicate$ empty$

```

```

1599     'skip$
1600     { "/" swap$ *
1601     }
1602     if$ *
1603   }
1604   if$ *
1605 }
1606 { duplicate$ "ECR" =
1607   { "journal" bibinfo.check
1608     volume "volume" bibinfo.check
1609     duplicate$ empty$
1610     'skip$
1611     { " " swap$ * "--" * * }
1612     if$
1613   }
1614   { "journal" bibinfo.check

```

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

```

1615     is.online
1616     { " " * bbl.online *
1617       #1 'online.shown :=
1618     }
1619     'skip$
1620     if$

```

```

1621     volume "volume" bibinfo.check
1622     duplicate$ empty$
1623     'pop$
1624     { " " * swap$ * }
1625     if$
1626     pages empty$
1627     eid empty$
1628     and
1629     'skip$
1630     { " " * }
1631     if$
1632   }
1633   if$
1634 }
1635 if$
1636 eid empty$
1637 { pages "pages" bibinfo.check * }
1638 { eid "eid" bibinfo.check * }
1639 if$
1640 }

```

7.6.7 Access and URLs

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

```

1641 FUNCTION {format.eprint}
1642 { eprint duplicate$ empty$
1643   'skip$
1644   { "\eprint"
1645     archive empty$
1646     'skip$
1647     { "[" * archive * "]" * }
1648     if$
1649     "{" * swap$ * "}" *
1650   }
1651   if$
1652 }

```

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

```

1653 FUNCTION {format.url}
1654 { doi empty$
1655   { url }
1656   { "https://doi.org/" doi * }

```

```

1657 if$
1658 duplicate$ empty$
1659 { pop$ "" }
1660 { "\urlprefix\url{" swap$ * "}" * }
1661 if$
1662 duplicate$ empty$
1663 urlyear empty$ not
1664 urldate empty$ not
1665 or
1666 and
1667 { library duplicate$ empty$
1668   'pop$
1669   { "\urlprefix{" swap$ * * }
1670   if$
1671   }
1672   'skip$
1673 if$
1674 urlyear empty$
1675 { urldate empty$
1676   'skip$
1677   { " [\urldateprefix{" * urldate * "]" * }
1678   if$
1679   }
1680   { " [\urldateprefix{" * urlyear * "]" * }
1681   if$
1682 }

```

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).

```

1683 FUNCTION {format.article.crossref}
1684 { word.in
1685   " \cite{" * crossref * "}" *
1686 }
1687 FUNCTION {format.incoll.inproc.crossref}
1688 { word.in
1689   " \cite{" * crossref * "}" *
1690 }

```

`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.

```

1691 FUNCTION {format.book.crossref}
1692 { volume duplicate$ empty$
1693   { "empty volume in " cite$ * "'s crossref of " * crossref * warning$
1694     pop$ word.in
1695   }
1696   { bbl.volume
1697     capitalize
1698     swap$ tie.or.space.prefix "volume" bibinfo.check * * bbl.of space.word *
1699   }
1700   if$
1701   " \cite{" * crossref * "}" *
1702 }

```

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

```

1704 FUNCTION {article}
1705 { output.bibitem
1706   format.authors "author" output.check
1707   author format.key output
1708   format.date "year" output.check

```

```

1709 date.block
1710 format.title "title" output.check
1711 print.title.addenda
1712 new.block
1713 crossref missing$
1714   { journal "t" change.case$
1715     "journal" bibinfo.check
1716     emphasize
1717     "journal" output.check

```

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

```

1718 is.online
1719   { continue.clause
1720     bbl.online output
1721     #1 'online.shown :=
1722   }
1723   'skip$
1724   if$

```

```

1725 format.vol.num.pages output
1726 }
1727 { format.article.crossref output.nonnull
1728   format.pages output
1729 }
1730 if$
1731 new.block
1732 format.note output
1733 new.block
1734 format.eprint output
1735 format.url output
1736 fin.entry
1737 }

```

7.7.2 Book

```

1738 FUNCTION {book}
1739 { output.bibitem
1740   author empty$
1741   { editor empty$
1742     type$ "reference" =
1743     or
1744     { type$ "reference" =
1745       { format.title "title" output.check }
1746       { format.btitle "title" output.check }
1747     if$
1748     print.labeltitle.addenda
1749     format.date "year" output.check
1750     year empty$ 'skip$ { date.block } if$
1751     print.title.addenda
1752   }
1753   { format.editors "author and editor" output.check
1754     format.date "year" output.check
1755     date.block
1756     format.btitle "title" output.check
1757     print.title.addenda
1758   }
1759   if$
1760 }
1761 { format.authors output.nonnull
1762   crossref missing$
1763   { "author and editor" editor either.or.check }
1764   'skip$
1765   if$
1766   format.date "year" output.check
1767   date.block
1768   format.btitle "title" output.check
1769   print.title.addenda
1770 }
1771 if$
1772 format.bvolume output

```



```

1773 format.number.series output
1774 new.block
1775 format.edition output
1776 new.block
1777 format.note output
1778 crossref missing$
1779   { keywords field.or.null "uksi" =
1780     { mid.sentence 'output.state := }
1781     { new.sentence }
1782     if$
1783     format.publisher.address output
1784   }
1785   { new.block
1786     format.book.crossref output.nonnull
1787   }
1788 if$
1789 new.block
1790 format.eprint output
1791 format.url output
1792 fin.entry
1793 }

```

7.7.3 Unpublished or self-published monograph

```

1794 FUNCTION {booklet}
1795 { output.bibitem
1796   author empty$
1797   { format.avtitle "title" output.check
1798     print.labeltitle.addenda
1799     format.date "year" output.check
1800     year empty$ 'skip$ { date.block } if$
1801     print.title.addenda
1802   }
1803   { format.authors output
1804     author format.key output
1805     format.date "year" output.check
1806     date.block
1807     format.avtitle "title" output.check
1808     print.title.addenda
1809   }
1810 if$
1811 new.block
1812 format.series.episode output

```

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

```

1813 is.online
1814   { continue.clause
1815     bbl.online output
1816     #1 'online.shown :=
1817   }
1818   'skip$
1819 if$

```

```

1820 new.block
1821 type "type" bibinfo.check output
1822 new.block
1823 howpublished "howpublished" bibinfo.check output
1824 new.block
1825 format.note output
1826 publisher empty$ 'skip$
1827   { new.block}
1828 if$
1829 format.publisher.address output
1830 new.block
1831 format.eprint output
1832 format.url output
1833 fin.entry
1834 }

```

7.7.4 Work in book

```
1835 FUNCTION {inbook}
1836 { output.bibitem
1837   author empty$
1838   { format.editors "author and editor" output.check
1839     editor format.key output
1840   }
1841   { format.authors output.nonnull
1842     crossref missing$
1843     { "author and editor" editor either.or.check }
1844     'skip$
1845     if$
1846   }
1847   if$
1848   format.date "year" output.check
1849   date.block
1850   format.btitle "title" output.check
1851   print.title.addenda
1852   new.block
1853   format.edition output
1854   new.block
1855   format.note output
1856   new.block
1857   crossref missing$
1858   { format.publisher.address output
1859     format.bvolume output
1860     format.chapter.pages "chapter and pages" output.check
1861     new.block
1862     format.number.series output
1863     new.sentence
1864   }
1865   { format.chapter.pages "chapter and pages" output.check
1866     new.block
1867     format.book.crossref output.nonnull
1868   }
1869   if$
1870   new.block
1871   format.eprint output
1872   format.url output
1873   fin.entry
1874 }
```

7.7.5 Work in collection

```
1875 FUNCTION {incollection}
1876 { output.bibitem
1877   author empty$
1878   { type$ "inreference" =
1879     { format.label.booktitle output}
1880     { format.title "title" output.check
1881       print.labeltitle.addenda
1882     }
1883     if$
1884     format.date "year" output.check
1885     year empty$ 'skip$ { date.block } if$
1886     type$ "inreference" =
1887     'skip$
1888     { print.title.addenda }
1889     if$
1890   }
1891   { format.authors output.nonnull
1892     format.date "year" output.check
1893     date.block
1894     format.title "title" output.check
1895     print.title.addenda
1896   }
1897   if$
1898   new.block
1899   crossref missing$
1900   { type$ "inreference" =
```

```

1901     author empty$
1902     and
1903     { format.byeditors output }
1904     { format.in.ed.booktitle "booktitle" output.check }
1905     if$
1906     new.block
1907     format.edition output
1908     new.block
1909     type$ "inreference" =
1910     author empty$
1911     and
1912     { format.btitle "title" output.check
1913       print.title.addenda
1914       new.block
1915     }
1916     'skip$
1917     if$
1918     format.note output
1919     new.block
1920     format.publisher.address output
1921     format.bvolume output
1922     format.number.series output
1923     format.chapter.pages output
1924   }
1925   { format.note output
1926     new.block
1927     format.incoll.inproc.crossref output.nonnull
1928     format.chapter.pages output
1929   }
1930   if$
1931   new.block
1932   format.eprint output
1933   format.url output
1934   fin.entry
1935 }

```

7.7.6 Conference paper

```

1936 FUNCTION {inproceedings}
1937 { output.bibitem
1938   format.authors "author" output.check
1939   author format.key output
1940   format.date "year" output.check
1941   date.block
1942   format.title "title" output.check
1943   print.title.addenda
1944   new.block
1945   crossref missing$
1946   { format.in.ed.booktitle "booktitle" output.check
1947     eventyear output
1948     venue output
1949     format.note output
1950     new.sentence
1951     publisher empty$
1952     { format.organization.address output }
1953     { organization "organization" bibinfo.check output
1954       format.publisher.address output
1955     }
1956     if$
1957     format.bvolume output
1958     format.number.series output
1959     format.pages output
1960   }
1961   { format.note output
1962     new.block
1963     format.incoll.inproc.crossref output.nonnull
1964     format.pages output
1965   }
1966   if$
1967   new.block
1968   format.eprint output

```

```

1969     format.url output
1970     fin.entry
1971   }
1972 FUNCTION {conference} { inproceedings }

```

7.7.7 Legal case studies

```

1973 FUNCTION {jurisdiction}
1974 { output.bibitem
1975   author empty$
1976   { format.btitle "title" output.check
1977     continue.clause
1978     format.case.number output
1979     number empty$
1980     casenumber empty$
1981     and
1982     'new.sentence
1983     'continue.clause
1984     if$
1985     print.labeltitle.addenda
1986     format.jur.date "year" output.check
1987     year empty$ 'skip$ { eu.case.check } if$
1988     print.title.addenda
1989   }
1990   { format.authors output
1991     author format.key output
1992     format.jur.date "year" output.check
1993     date.block
1994     continue.clause
1995     format.btitle "title" output.check
1996     print.labeltitle.addenda
1997     print.title.addenda
1998   }
1999   if$
2000   eu.case.check
2001   format.note output
2002   note empty$
2003   'eu.case.check
2004   'new.block
2005   if$
2006   crossref missing$
2007   { continue.clause
2008     format.journal.series.vol.pages output
2009   }
2010   { format.article.crossref output.nonnull
2011     format.pages output
2012   }
2013   if$
2014   new.block
2015   format.eprint output
2016   format.url output
2017   fin.entry
2018 }

```

7.7.8 Legislation

```

2019 FUNCTION {uklegislation}
2020 { output.bibitem
2021   author empty$
2022   { format.btitle "title" output.check
2023     print.labeltitle.addenda
2024     continue.clause
2025     format.date emphasize "year" output.check
2026     print.title.addenda
2027   }
2028   { format.authors output
2029     author format.key output
2030     format.date "year" output.check
2031     date.block

```

```

2032     continue.clause
2033     format.btitle "title" output.check
2034     print.labeltitle.addenda
2035     print.title.addenda
2036   }
2037   if$
2038   entrysubtype field.or.null "secondary" =
2039     { number "number" bibinfo.check output }
2040     { series empty$ type empty$ and
2041       { number empty$
2042         { format.chapter output }
2043         { continue.clause
2044           format.series.number.chapter output
2045         }
2046       }
2047     }
2048     { chapter empty$
2049       { new.block }
2050       { continue.clause }
2051     }
2052     if$
2053     format.series.number.chapter output
2054   }
2055 }
2056 if$
2057 note empty$
2058 'skip$
2059 { new.block
2060   format.note output
2061   new.block
2062 }
2063 if$
2064 publisher empty$ 'skip$
2065 { entrysubtype field.or.null "secondary" =
2066   'skip$
2067   { new.block }
2068   if$
2069   format.publisher.address output
2070 }
2071 if$
2072 new.block
2073 format.eprint output
2074 format.url output
2075 fin.entry
2076 }
2077 FUNCTION {eulegislation}
2078 { output.bibitem
2079   author empty$
2080   { format.title "title" output.check
2081     print.labeltitle.addenda
2082     continue.clause
2083     format.eu.date "year" output.check
2084     print.title.addenda
2085   }
2086   { format.authors output
2087     author format.key output
2088     format.eu.date "year" output.check
2089     date.block
2090     continue.clause
2091     format.title "title" output.check
2092     print.labeltitle.addenda
2093     print.title.addenda
2094   }
2095   if$
2096   eu.case.check
2097   format.note output
2098   note empty$
2099   'eu.case.check
2100   'new.block
2101   if$
2102   crossref missing$
2103   { continue.clause
2104     format.journal.series.vol.pages output

```

```

2105     }
2106     { format.article.crossref output.nonnull
2107       format.pages output
2108     }
2109     if$
2110     new.block
2111     format.eprint output
2112     format.url output
2113     fin.entry
2114   }
2115   FUNCTION {legislation}
2116   { journal empty$
2117     'uklegislation
2118     'eulegislation
2119     if$
2120   }

```

7.7.9 Manual

```

2121   FUNCTION {manual}
2122   { output.bibitem
2123     author empty$
2124     { format.btitle "title" output.check
2125       print.labeltitle.addenda
2126       format.date "year" output.check
2127       year empty$ 'skip$ { date.block } if$
2128       print.title.addenda
2129     }
2130     { format.authors output
2131       author format.key output
2132       format.date "year" output.check
2133       date.block
2134       format.btitle "title" output.check
2135       print.title.addenda
2136     }
2137     if$
2138     month "month" bibinfo.check output
2139     format.edition output
2140     new.block
2141     format.manual.number output
2142     new.block
2143     format.note output
2144     new.block
2145     format.library output
2146     organization address new.block.checkb
2147     format.organization.address output
2148     new.block
2149     format.eprint output
2150     format.url output
2151     fin.entry
2152   }

```

7.7.10 Image

```

2153   FUNCTION {image}
2154   { output.bibitem
2155     author empty$
2156     { format.btitle "title" output.check
2157       print.labeltitle.addenda
2158       format.date "year" output.check
2159       year empty$ 'skip$ { date.block } if$
2160       print.title.addenda
2161     }
2162     { format.authors output
2163       author format.key output
2164       format.date "year" output.check
2165       date.block
2166       format.btitle "title" output.check
2167       print.title.addenda
2168     }

```

```

2169     if$
2170     format.edition output
2171     new.block
2172     format.note output
2173     new.block
2174     format.manual.number output
2175     new.block
2176     format.pub.org.lib.address output
2177     new.block
2178     format.eprint output
2179     format.url output
2180     fin.entry
2181 }

```

7.7.11 Master's thesis

```

2182 FUNCTION {mastersthesis}
2183 { output.bibitem
2184   format.authors "author" output.check
2185   author format.key output
2186   format.date "year" output.check
2187   date.block
2188   format.btitle "title" output.check
2189   print.title.addenda
2190   new.block
2191   format.note output
2192   new.block
2193   bbl.mthesis format.thesis.type output.nonnull
2194   new.block
2195   school "school" bibinfo.warn output
2196   address "address" bibinfo.check output
2197   new.block
2198   format.eprint output
2199   format.url output
2200   fin.entry
2201 }

```

7.7.12 Miscellaneous

```

2202 FUNCTION {misc}
2203 { output.bibitem
2204   author empty$
2205   { format.title "title" output.check
2206     print.labeltitle.addenda
2207     format.date "year" output.check
2208     year empty$ 'skip$ { date.block } if$
2209     print.title.addenda
2210   }
2211   { format.authors output
2212     author format.key output
2213     format.date "year" output.check
2214     date.block
2215     format.title "title" output.check
2216     print.title.addenda
2217   }
2218   if$
2219   new.block
2220   howpublished "howpublished" bibinfo.check output
2221   new.block
2222   format.note output
2223   new.block
2224   format.eprint output
2225   format.url output
2226   fin.entry
2227 }

```

7.7.13 Doctoral thesis

```
2228 FUNCTION {phdthesis}
2229 { output.bibitem
2230   format.authors "author" output.check
2231   author format.key output
2232   format.date "year" output.check
2233   date.block
2234   format.btitle "title" output.check
2235   print.title.addenda
2236   new.block
2237   format.note output
2238   new.block
2239   bbl.phdthesis format.thesis.type output.nonnull
2240   new.block
2241   school "school" bibinfo.warn output
2242   address "address" bibinfo.check output
2243   new.block
2244   format.eprint output
2245   format.url output
2246   fin.entry
2247 }
```

7.7.14 Conference proceedings

```
2248 FUNCTION {proceedings}
2249 { output.bibitem
2250   editor empty$
2251   { format.btitle "title" output.check
2252     print.labeltitle.addenda
2253     format.date "year" output.check
2254     year empty$ 'skip$ { date.block } if$
2255     print.title.addenda
2256   }
2257   { format.editors output
2258     editor format.key output
2259     format.date "year" output.check
2260     date.block
2261     format.btitle "title" output.check
2262     print.title.addenda
2263   }
2264   if$
2265   format.bvolume output
2266   format.number.series output
2267   format.note output
2268   new.sentence
2269   publisher empty$
2270   { format.organization.address output }
2271   { organization "organization" bibinfo.check output
2272     format.publisher.address output
2273   }
2274   if$
2275   new.block
2276   format.eprint output
2277   format.url output
2278   fin.entry
2279 }
```

7.7.15 Report

```
2280 FUNCTION {techreport}
2281 { output.bibitem
2282   author empty$
2283   { format.btitle "title" output.check
2284     print.labeltitle.addenda
2285     format.date "year" output.check
2286     year empty$ 'skip$ { date.block } if$
2287     print.title.addenda
2288   }
```



```

2289     { format.authors output
2290       author format.key output
2291       format.date "year" output.check
2292       date.block
2293       format.btitle "title" output.check
2294       print.title.addenda
2295     }
2296     if$
2297     new.block
2298     format.tr.number output
2299     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`:

```

2300     format.note output
2301     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):

```

2160     library empty$
2161     'skip$
2162     { format.note output
2163       new.block
2164     }
2165     if$

```

This bit is the same.

```

2302     publisher empty$
2303     { format.institution.address output }
2304     { institution "institution" bibinfo.check output
2305       format.publisher.address output
2306     }
2307     if$
2308     new.block

```

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

```

2309     format.library output

```

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

```

2173     library empty$
2174     { format.note output }
2175     { format.library output }
2176     if$

```

The rest is the same.

```

2310     new.block
2311     format.eprint output
2312     format.url output
2313     fin.entry
2314 }

```

7.7.16 Unpublished works

```

2315 FUNCTION {unpublished}
2316 { output.bibitem
2317   author empty$
2318   { booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2319     print.labeltitle.addenda
2320     format.date "year" output.check
2321     year empty$ 'skip$ { date.block } if$
2322     print.title.addenda
2323   }
2324   { format.authors output
2325     author format.key output
2326     format.date "year" output.check
2327     date.block

```

```

2328     booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2329     print.title.addenda
2330   }
2331   if$
2332   new.block
2333   format.in.ed.booktitle output
2334   new.block
2335   howpublished "howpublished" bibinfo.check output
2336   new.block
2337   format.note output
2338   new.block
2339   bbl.unpublished output
2340   new.block
2341   format.eprint output
2342   format.url output
2343   fin.entry
2344 }

```

7.7.17 Aliases and fallbacks

Here are a list of aliases supported by this style.

```

2346 FUNCTION {letter}
2347 { journal empty$
2348   'unpublished
2349   'article
2350   if$
2351 }
2352 FUNCTION {collection} { book }
2353 FUNCTION {reference} { book }
2354 FUNCTION {inreference} { incollection }
2355 FUNCTION {audio} { booklet }
2356 FUNCTION {movie} { booklet }
2357 FUNCTION {music} { booklet }
2358 FUNCTION {video} { booklet }
2359 FUNCTION {dataset} { manual }
2360 FUNCTION {electronic} { manual }
2361 FUNCTION {online} { manual }
2362 FUNCTION {patent} { manual }
2363 FUNCTION {software} { manual }
2364 FUNCTION {standard} { manual }
2365 FUNCTION {www} { manual }
2366 FUNCTION {thesis} { phdthesis }
2367 FUNCTION {report} { techreport }

```

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

```

2368 FUNCTION {default.type} { misc }

```

7.8 Writing out the bibliography to a file

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

```

2370 READ

```

These functions are used to generate a sort key for each entry in the `.bib` file. `sortify` strips out \LaTeX commands, spaces and ASCII symbols, and converts to lowercase.

```

2371 FUNCTION {sortify}
2372 { purify$
2373   "l" change.case$
2374 }

```

`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`.

```

2375 INTEGERS { len }
2376 FUNCTION {chop.word}
2377 { 's :=
2378   'len :=

```

```

2379 s #1 len substring$ =
2380 { s len #1 + global.max$ substring$ }
2381 's
2382 if$
2383 }

```

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.'

```

2384 FUNCTION {format.lab.names}
2385 {'s :=
2386 "" 't :=
2387 #1 'nameptr :=
2388 s num.names$ 'numnames :=
2389 numnames 'namesleft :=
2390 { namesleft #0 > }
2391 { s nameptr
2392 "{vv~}{ll}" format.name$
2393 't :=
2394 nameptr #1 >
2395 { nameptr #2 =
2396 numnames #3 > and
2397 { "others" 't :=
2398 #1 'namesleft := }
2399 'skip$
2400 if$
2401 namesleft #1 >
2402 { ", " * t * }
2403 { s nameptr "{ll}" format.name$ duplicate$ "others" =
2404 { 't := }
2405 { pop$ }
2406 if$
2407 t "others" =
2408 { " " * bbl.etal *
2409 }
2410 { bbl.and
2411 space.word * t *
2412 }
2413 if$
2414 }
2415 if$
2416 }
2417 't
2418 if$
2419 nameptr #1 + 'nameptr :=
2420 namesleft #1 - 'namesleft :=
2421 }
2422 while$
2423 }

```

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

```

2424 FUNCTION {author.editor.key.label}
2425 { author empty$
2426 { editor empty$
2427 { key empty$
2428 { title empty$
2429 { cite$ #1 #3 substring$ }
2430 { title
2431 type$ "book" =
2432 { emphasize }
2433 'skip$
2434 if$
2435 }
2436 if$
2437 }
2438 'key
2439 if$
2440 }
2441 { editor format.lab.names }
2442 if$
2443 }

```

```

2444     { author format.lab.names }
2445     if$
2446 }

```

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

```

2447 FUNCTION {editor.key.label}
2448 { editor empty$
2449   { key empty$
2450     { title empty$
2451       { cite$ #1 #3 substring$ }
2452       { title emphasize }
2453       if$
2454     }
2455     'key
2456     if$
2457   }
2458   { editor format.lab.names }
2459   if$
2460 }

```

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

```

2461 FUNCTION {author.key.label}
2462 { author empty$
2463   { key empty$
2464     { type$ "inreference" =
2465       booktitle empty$ not
2466       and
2467       { booktitle }
2468       { title empty$
2469         { cite$ #1 #3 substring$ }
2470         { title
2471           type$ "article" =
2472           type$ "incollection" =
2473           or
2474           type$ "inproceedings" =
2475           or
2476           type$ "reference" =
2477           author empty$
2478           and
2479           or
2480           type$ "misc" =
2481           or
2482           'skip$
2483           { emphasize }
2484           if$
2485         }
2486         if$
2487       }
2488     }
2489     'key
2490     if$
2491   }
2492   { author format.lab.names }
2493   if$
2494 }
2495 }

```

calc.short.authors chooses between the above.

```

2496 FUNCTION {calc.short.authors}
2497 { type$ "book" =
2498   type$ "inbook" =
2499   or
2500   'author.editor.key.label
2501   { type$ "proceedings" =
2502     'editor.key.label
2503     'author.key.label
2504     if$
2505   }

```

```

2506   if$
2507   'short.list :=
2508 }

```

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.

```

2509 FUNCTION {calc.label}
2510 { calc.short.authors
2511   short.list
2512   "("
2513   *
2514   year nodate.check duplicate$ empty$
2515     { pop$ "" }
2516     'skip$
2517   if$
2518   *
2519   'label :=
2520 }

```

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.’

```

2521 FUNCTION {sort.format.names}
2522 { 's :=
2523   #1 'nameptr :=
2524   ""
2525   s num.names$ 'numnames :=
2526   numnames 'namesleft :=
2527   { namesleft #0 > }
2528   { s nameptr
2529     "{ll}{ }{ f }{ }{ jj }{ }{ vv }{"
2530     format.name$ 't :=
2531     nameptr #1 >
2532     { " " *
2533       namesleft #1 =
2534       t "others" =
2535       and
2536       { "zzzzz" 't := }
2537       'skip$
2538     if$
2539     t sortify *
2540   }
2541   { t sortify * }
2542   if$
2543   nameptr #1 + 'nameptr :=
2544   namesleft #1 - 'namesleft :=
2545   }
2546   while$
2547 }

```

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

```

2548 FUNCTION {sort.format.title}
2549 { 't :=
2550   "A " #2
2551   "An " #3
2552   "The " #4 t chop.word
2553   chop.word
2554   chop.word
2555   sortify
2556   #1 global.max$ substring$
2557 }

```

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.

```

2558 FUNCTION {author.editor.sort}
2559 { author empty$
2560   { editor empty$
2561     { title empty$
2562       { key empty$
2563         { "to sort, need author, editor, title, or key in " cite$ * warning$
2564           ""
2565         }
2566         { key sortify }
2567         if$
2568       }
2569       { title sort.format.title }
2570       if$
2571     }
2572     { editor sort.format.names }
2573     if$
2574   }
2575   { author sort.format.names }
2576   if$
2577 }

```

editor.sort does similar but skips the author names.

```

2578 FUNCTION {editor.sort}
2579 { editor empty$
2580   { title empty$
2581     { key empty$
2582       { "to sort, need editor, title, or key in " cite$ * warning$
2583         ""
2584       }
2585       { key sortify }
2586       if$
2587     }
2588     { title sort.format.title }
2589     if$
2590   }
2591   { editor sort.format.names }
2592   if$
2593 }

```

author.sort does similar but skips the editor names.

```

2594 FUNCTION {author.sort}
2595 { author empty$
2596   { type$ "inreference" =
2597     booktitle empty$ not
2598     and
2599     { booktitle sort.format.title }
2600     { title empty$
2601       { key empty$
2602         { "to sort, need author, title, or key in " cite$ * warning$
2603           ""
2604         }
2605         { key sortify }
2606         if$
2607       }
2608       { title sort.format.title }
2609       if$
2610     }
2611     if$
2612   }
2613   { author sort.format.names }
2614   if$
2615 }

```

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.

```

2617 FUNCTION {presort}
2618 { calc.label
2619   label sortify
2620   " "
2621   *

```

```

2622 type$ "book" =
2623 type$ "inbook" =
2624 or
2625   'author.editor.sort
2626   { type$ "proceedings" =
2627     'editor.sort
2628     'author.sort
2629     if$
2630   }
2631 if$
2632 #1 entry.max$ substring$
2633 'sort.label :=
2634 sort.label
2635 *
2636 " "
2637 *
2638 title field.or.null
2639 sort.format.title
2640 *
2641 #1 entry.max$ substring$
2642 'sort.key$ :=
2643 }

```

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.

```

2645 ITERATE {presort}
2646 SORT

```

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

```

2647 STRINGS { last.label next.extra }
2648 INTEGERS { last.extra.num last.extra.num.extended last.extra.num.blank number.label }
2649 FUNCTION {initialize.extra.label.stuff}
2650 { #0 int.to.chr$ 'last.label :=
2651   "" 'next.extra :=
2652   #0 'last.extra.num :=
2653   "a" chr.to.int$ #1 - 'last.extra.num.blank :=
2654   last.extra.num.blank 'last.extra.num.extended :=
2655   #0 'number.label :=
2656 }
2657 FUNCTION {forward.pass}
2658 { last.label label =
2659   { last.extra.num #1 + 'last.extra.num :=
2660     last.extra.num "z" chr.to.int$ >
2661     { "a" chr.to.int$ 'last.extra.num :=
2662       last.extra.num.extended #1 + 'last.extra.num.extended :=
2663     }
2664     'skip$
2665   if$
2666   last.extra.num.extended last.extra.num.blank >
2667   { last.extra.num.extended int.to.chr$
2668     last.extra.num int.to.chr$
2669     * 'extra.label := }
2670   { last.extra.num int.to.chr$ 'extra.label := }
2671   if$
2672 }
2673 { "a" chr.to.int$ 'last.extra.num :=
2674   "" 'extra.label :=
2675   label 'last.label :=
2676 }
2677 if$
2678 number.label #1 + 'number.label :=
2679 }
2680 FUNCTION {reverse.pass}
2681 { next.extra "b" =
2682   { "a" 'extra.label := }
2683   'skip$
2684 if$
2685 extra.label 'next.extra :=
2686 extra.label
2687 duplicate$ empty$

```

```

2688     'skip$
2689     { "\natexlab{" swap$ * "}" * }
2690     if$
2691     'extra.label :=
2692     label extra.label * 'label :=
2693 }

```

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.

```

2694 EXECUTE {initialize.extra.label.stuff}
2695 ITERATE {forward.pass}
2696 REVERSE {reverse.pass}
2697 FUNCTION {bib.sort.order}
2698 { sort.label
2699   "   "
2700   *
2701   year field.or.null sortify
2702   *
2703   "   "
2704   *
2705   title field.or.null
2706   sort.format.title
2707   *
2708   #1 entry.max$ substring$
2709   'sort.key$ :=
2710 }
2711 ITERATE {bib.sort.order}
2712 SORT

```

We now write out the beginning of the bibliography to a `.bbl` file. This includes opening a \LaTeX environment, `thebibliography`, and some user-redefinable strings.

```

2713 FUNCTION {begin.bib}
2714 { preamble$ empty$
2715   'skip$
2716   { preamble$ write$ newline$ }
2717   if$
2718   "\begin{thebibliography}{ number.label int.to.str$ * "}" *
2719   write$ newline$
2720   "\providecommand{\natexlab}[1]{#1}"
2721   write$ newline$
2722   "\providecommand{\url}[1]{\texttt{#1}}"
2723   write$ newline$
2724   "\providecommand{\urlprefix}{Available from: }"
2725   write$ newline$
2726   "\providecommand{\urldateprefix}{Accessed }"
2727   write$ newline$
2728   "\providecommand{\selectlanguage}[1]{\relax}"
2729   write$ newline$
2730   "\providecommand{\bibinfo}[2]{#2}"
2731   write$ newline$
2732   "\providecommand{\eprint}[2][ ]{\url{#2}}"
2733   write$ newline$
2734 }
2735 EXECUTE {begin.bib}
2736 EXECUTE {init.state.consts}

```

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

```

2737 ITERATE {call.type$}

```

After the last entry, we close the \LaTeX environment and stop writing to the `.bbl` file.

```

2738 FUNCTION {end.bib}
2739 { newline$
2740   "\end{thebibliography}" write$ newline$
2741 }
2742 EXECUTE {end.bib}

```