

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

Maintainer: Alex Ball*

Package v5.0 – 4 March 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.


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_T_EX. The last shows how the reference was entered in the `.bib` file. The bottom right corner shows the source of the example: ‘RX’ indicates the ‘Reference examples (A-Z)’ section of the Guide; ‘RL’ indicates the ‘Organise a reference list’ section of the Guide; ‘RI’ indicates *Referencing images*.

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

5.1 Books and book chapters

Book with author(s)


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

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

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

RX

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

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

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

RX

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

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

RX

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

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

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

RX

- 📖 Internally, collection is an alias for book.

Electronic book

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

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

RX

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

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

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

RX

Book known by its title

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

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

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

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

RX

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

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

⚙️ 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 BibTeX and biblatex, one unfortunate difference is that you must escape the hash symbol inside URLs with BibTeX but you must not escape it inside a URL with biblatex.

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

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

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

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

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

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

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

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

RL

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

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

RX

5.2 Articles and periodicals

Journal article

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

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

RL

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

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

RL

- 📖 Newman, R., 2010. Malaria control beyond 2010. *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 Ostrovnaya, I., 2021. Identifying prognostic pairwise relationships among bacterial species in microbiome studies. *PLOS Computational Biology* [Online], 17(11). Available from: <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501> [Accessed 9 December 2021].
- Devlin, S.M., Martin, A. and Ostrovnaya, I., 2021. Identifying prognostic pairwise relationships among bacterial species in microbiome studies. *PLOS Computational Biology* [Online], 17(11). Available from: <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501> [Accessed 9 December 2021].

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

RX

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

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

RX

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

- 📖 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

- 💡 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.

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_TE_X, but are needed if you want compatibility with bib_lat_ex.

- 📖 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

 Anon., 1946. *Prototype electric bicycle displayed at the ‘Britain Can Make It’ exhibition organised by the Council of Industrial Design and held at the Victoria and Albert Museum, London, 1946. Designed by B. G. Bowden* [Photograph]. At: London. Design Council Slide Collection.

 Anon., 1946. *Prototype electric bicycle displayed at the ‘Britain Can Make It’ exhibition organised by the Council of Industrial Design and held at the Victoria and Albert Museum, London, 1946. Designed by B. G. Bowden* [Photograph]. At: London. Design Council Slide Collection.



```
@image{anon1946peb,  
  author = {Anon.},  
  year = {1946},  
  title = {Prototype electric bicycle displayed at the \enquote{Britain Can Make It} exhibition  
    organised by the {Council} of {Industrial} {Design} and held at the {Victoria} and {Albert}  
    {Museum}, {London}, 1946. {Designed} by {B.~G.} {Bowden}},  
  titleaddon = {Photograph},  
  address = {London},  
  library = {Design Council Slide Collection}}
```

RI

 Use the **library** field to record the archive and register number. 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.

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},  
  titleaddon = {computer program},  
  url = {https://screencast-o-matic.com/},  
  urlyear = {16 May 2016}}
```

RX

❗ Internally, `software` is an alias for `manual`.

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

```
@article{thurfjell1975vhv,  
  author = {Thurfjell, W.},  
  year = {1975},  
  title = {Vart har vårän 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 \textup{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:

<pre>\defcitealias{eu.dir1984/2003}{% Council Regulation [EC] 1984/2003} \citealias{eu.dir1984/2003}</pre>	(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.
- Anon., 1946. *Prototype electric bicycle displayed at the 'Britain Can Make It' exhibition organised by the Council of Industrial Design and held at the Victoria and Albert Museum, London, 1946. Designed by B. G. Bowden* [Photograph]. At: London. Design Council Slide Collection.

- 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.
- 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) 1984/2003 of 8 April 2003 introducing a system for the statistical monitoring of trade in bluefin tuna, swordfish and big eye tuna within the Community [2003] *OJ* L295.
- Crawford, G.I., 1965. Oxygen in metals. In: J.M.A. Lenihan and S.J. Thompson, eds. *Activation analysis: Proceedings of a NATO Advanced Study Institute*, 2–4 August 1964, Glasgow. London: Academic Press, pp.113–118.
- Deneulin, S. and Dinerstein, A.C., 2010. *Hope movements: Social movements in the pursuit of human development*. (Bath papers in international development and wellbeing, no. 8). Bath: University of Bath.
- Devlin, S.M., Martin, A. and Ostrovnya, I., 2021. Identifying prognostic pairwise relationships among bacterial species in microbiome studies. *PLOS Computational Biology* [Online], 17(11). Available from: <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1009501> [Accessed 9 December 2021].
- The elephant man* [Online], 1980. Film. Directed by David Lynch. USA: Brookfilms. Available from: BBC iPlayer [Accessed 4 May 2021].

- Esquivel, L., 2003. *Como agua para chocolate* [Like water for chocolate]. Barcelona: Debolsillo.
- European Commission, 2015. *General report on the activities of the European Union 2014*. Luxembourg: Publications Office of the European Union.
- Gaiman, N., 2021. *Facebook post* [Online], 21 April. Available from: <https://www.facebook.com/neilgaiman/posts/305187897642814> [Accessed 13 May 2021].
- Ganju, V., 2021. *A study of EnGeneIC Dream Vectors (EDV's) packaged with the chemotherapy, E-EDV-D682 given simultaneously as non-targeted EDVs carrying an immune enhancer called EDV-GC, in participants with advanced pancreatic and other cancers whose disease has progressed after one or two treatment regimes, or where other standard therapies are not appropriate*. Sydney: University of Sydney. *Australian and New Zealand Clinical Trials Registry* [Online]. Available from: <http://www.anzctr.org.au/Trial/Registration/TrialReview.aspx?id=365258> [Accessed 29 April 2021].
- Google, 2020. *Harbourside, Bristol* [Online], Google Maps. Available from: <https://www.google.co.uk/maps/place/Harbourside,+Bristol/> [Accessed 30 April 2020].
- Great Britain. Ministry of Defence, 2004. *Delivering security in a changing world: defence white paper*. (Cm. 6041). London: TSO.
- Great Britain. Parliament. House of Commons, 1988. *Local government finance bill*. (Bills | 1987/88, 66). London: HMSO.
- Great Britain. Parliament. House of Commons, 2004. *National Savings investment deposits: account 2002–2003*. (HC 2003/04, 30). London: National Audit Office.
- Great Britain. Parliament. House of 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.hbcpnetbase.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 vsekidnevnie 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., 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.
- 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 `bathx.bst`.

7.1 Data model

We begin by listing the entry fields that we will recognize. These are the standard Bib_T_E_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 references out to a file

We define a set of integers for saving an output state. We will use them to 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 removes an element from the stack, writes the appropriate delimiter to the `.bbl` file depending on the state, then restores the original element. Afterwards, the output state is set to `mid.sentence`.

- `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.

```

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       }
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 }

```

Before writing anything out, we need to make sure there is something to be printed. We provide two functions for this: `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. If the value is non-null, the tokens are concatenated in the form `\bibinfo{field name}{field value}`. Otherwise, the field name is removed from the stack; if the value is missing rather than empty, it is replaced with an empty string. 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.

```
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 allows us to convert cardinal numbers to English ordinals.

```

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 }

```


7.5 Name handling

`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.

```

411 INTEGERS { nameptr namesleft numnames }
412 STRINGS { bibinfo }
413 FUNCTION {format.names}
414 { 'bibinfo :=
415   duplicate$ empty$
416   'skip$
417   { 's :=
418     "" 't :=
419     #1 'nameptr :=
420     s num.names$ 'numnames :=
421     numnames 'namesleft :=
422     { namesleft #0 > }
423     { s nameptr
424       "{ll}{, jj}{, f{.}.}{~vv}"
425       format.name$
426       bibinfo bibinfo.check
427       't :=
428       nameptr #1 >
429       { namesleft #1 >
430         { ", " * t * }
431         { s nameptr "{ll}" format.name$ duplicate$ "others" =
432           { 't := }
433           { pop$ }
434           if$
435           t "others" =
436           { " " * bbl.etal *
437             }
438           { bbl.and
439             space.word * t *
440             }
441           if$
442           }
443           if$
444           }
445           't
446           if$
447           nameptr #1 + 'nameptr :=
448           namesleft #1 - 'namesleft :=
449           }
450         while$
451       }
452     if$
453   }

```

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

```

454 FUNCTION {format.names.ed}
455 { 'bibinfo :=
456   duplicate$ empty$
457   'skip$
458   { 's :=
459     "" 't :=
460     #1 'nameptr :=
461     s num.names$ 'numnames :=
462     numnames 'namesleft :=
463     { namesleft #0 > }
464     { s nameptr
465       "{f{.}.~}{vv~}{ll}{ jj}"
466       format.name$
467       bibinfo bibinfo.check
468       't :=
469       nameptr #1 >
470       { namesleft #1 >
471         { ", " * t * }
472         { s nameptr "{ll}" format.name$ duplicate$ "others" =
473           { 't := }

```

```

474         { pop$ }
475         if$
476         t "others" =
477         { " " * bbl.etal *
478         }
479         { bbl.and
480         space.word * t *
481         }
482         if$
483     }
484     if$
485 }
486 't
487 if$
488 nameptr #1 + 'nameptr :=
489 namesleft #1 - 'namesleft :=
490 }
491 while$
492 }
493 if$
494 }

```

`format.authors` returns author names.

```

495 FUNCTION {format.authors}
496 { author "author" format.names
497 }

```

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

```

498 FUNCTION {format.byeditors}
499 { editor "editor" format.names.ed duplicate$ empty$ 'skip$
500   { bbl.byeditor
501     " " *
502     swap$ *
503   }
504   if$
505 }

```

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

```

506 FUNCTION {get.bbl.editor}
507 { editor num.names$ #1 > 'bbl.editors 'bbl.editor if$ }
508 FUNCTION {format.editors}
509 { editor "editor" format.names duplicate$ empty$ 'skip$
510   { ", " *
511     " " *
512     get.bbl.editor
513     *
514   }
515   if$
516 }

```

`format.translators` does likewise for translators, in the form ‘(name, Trans.)’.

```

517 FUNCTION {format.translators}
518 { translator "translator" format.names.ed duplicate$ empty$ 'skip$
519   { ", " *
520     " " *
521     bbl.translator *
522     "(" swap$ *
523     ")" *
524   }
525   if$
526 }

```

`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.

```

527 FUNCTION {format.full.names}
528 { 's :=
529   " " 't :=
530   #1 'nameptr :=

```

```

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

```

`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.

```

561 FUNCTION {author.editor.key.full}
562 { author empty$
563   { editor empty$
564     { key empty$
565       { cite$ #1 #3 substring$ }
566       'key
567       if$
568     }
569     { editor format.full.names }
570     if$
571   }
572   { author format.full.names }
573   if$
574 }

```

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

```

575 FUNCTION {editor.key.full}
576 { editor empty$
577   { key empty$
578     { cite$ #1 #3 substring$ }
579     'key
580     if$
581   }
582   { editor format.full.names }
583   if$
584 }

```

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

```

585 FUNCTION {author.key.full}
586 { author empty$
587   { key empty$
588     { cite$ #1 #3 substring$ }
589     'key
590     if$
591   }

```

```

592 { author format.full.names }
593 if$
594 }

```

make.full.names chooses between the above.

```

595 FUNCTION {make.full.names}
596 { type$ "book" =
597   type$ "inbook" =
598   or
599   'author.editor.key.full
600   { type$ "proceedings" =
601     'editor.key.full
602     'author.key.full
603     if$
604   }
605   if$
606 }

```

output.bibitem writes out the very start of the 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 and puts the entry driver into the `before.all` output state.

```

607 FUNCTION {output.bibitem}
608 { newline$
609   "\bibitem[{" write$
610   label write$
611   ")" make.full.names duplicate$ short.list =
612   { pop$ }
613   { * }
614   if$
615   "]}{" * write$
616   cite$ write$
617   "]" write$
618   newline$
619   ""
620   #0 'online.shown :=
621   before.all 'output.state :=
622 }

```

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.

```

623 FUNCTION {format.key}
624 { empty$
625   { key field.or.null }
626   { "" }
627   if$
628   nameaddon duplicate$ empty$
629   'pop$
630   { swap$ duplicate$ empty$
631     'skip$
632     { " " * }
633     if$
634     swap$ "[" swap$ * * "]" *
635     continue.clause
636   }
637   if$
638 }

```

7.6 Titles

`select.language` checks if the field before was written out, and if so, wraps `{\selectlanguage{language}...}` around it.

```

639 FUNCTION {select.language}
640 { duplicate$ empty$
641   'pop$
642   { language empty$
643     'skip$
644     { "\selectlanguage{" language * "}" * swap$ * "}" * }
645     if$
646   }
647   if$
648 }

```

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

```

649 FUNCTION {format.title}
650 { title
651   duplicate$ empty$ 'skip$
652   { "t" change.case$ }
653   if$
654   "title" bibinfo.check
655   duplicate$ empty$
656   'skip$
657   { select.language }
658   if$
659 }

```

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

```

660 FUNCTION {format.btitle}
661 { title
662   duplicate$ empty$ 'skip$
663   { "t" change.case$ }
664   if$
665   "title" bibinfo.check
666   duplicate$ empty$
667   'skip$
668   { emphasize select.language }
669   if$
670 }

```

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

```

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

```

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

```

691 FUNCTION {format.titleaddon}
692 { titleaddon
693   duplicate$ empty$ 'skip$
694   { "titleaddon" bibinfo.check }
695   if$
696 }

```

This is the `bathx.bst` version:

```
691 FUNCTION {format.titleaddon}
692 { titleaddon
693   duplicate$ empty$ 'skip$
694   { "[" swap$ "titleaddon" bibinfo.check * "]" * }
695   if$
696 }
```

`format.version` returns the version field.

```
697 FUNCTION {format.version}
698 { version
699   duplicate$ empty$ 'skip$
700   { "(v." swap$ *
701     "version" bibinfo.check
702     ")" *
703   }
704   if$
705 }
```

`get.title.online` is a the function for writing '[Online]' after a title. In `bath.bst` it prints the `titleaddon` field:

```
706 FUNCTION {get.title.online}
707 { online.shown
708   'skip$
709   { continue.clause
710     format.titleaddon output
711     #1 'online.shown :=
712   }
713   if$
714 }
```

In `bathx.bst` it conditionally prints the `bbl.online` string:

```
706 FUNCTION {get.title.online}
707 { is.online
708   journal empty$
709   and
710   booktitle empty$
711   type$ "inreference" =
712   author empty$
713   and
714   or
715   and
716   library empty$
717   type$ duplicate$ "image" =
718   swap$ duplicate$ "booklet" =
719   swap$ duplicate$ "audio" =
720   swap$ duplicate$ "video" =
721   swap$ duplicate$ "music" =
722   swap$ "movie" =
723   or
724   or
725   or
726   or
727   or
728   or
729   and
730   type$ "software" = not
731   and
732   series empty$
733   type duplicate$ empty$
734   { pop$ #1 }
735   { duplicate$ "TV" =
736     swap$ "Radio" =
737     or not
738   }
739   if$
740   or
741   and
742   { continue.clause
743     bbl.online output
```

```

744     #1 'online.shown :=
745   }
746   'skip$
747   if$
748 }

```

`get.title.addenda` is a convenience for writing out the elements that normally follow the title in `mid.clause` state.

```

749 FUNCTION {get.title.addenda}
750 { continue.clause
751   format.version output

```

These two lines are peculiar to `bathx.bst`, where `titleaddon` doesn't get moved:

```

752 continue.clause
753 format.titleaddon output

```

```

754 get.title.online
755 continue.clause
756 type$ "reference" =
757   { format.byeditors output }
758   'skip$
759   if$
760   format.translators output
761 }

```

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

```

762 FUNCTION {word.in}
763 { bbl.in capitalize
764   ":" *
765   " " * }

```

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

```

766 FUNCTION {format.label.booktitle}
767 { booktitle
768   duplicate$ empty$ 'skip$
769   { "t" change.case$ }
770   if$
771   "booktitle" bibinfo.check
772 }

```

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

```

773 FUNCTION {format.booktitle}
774 { format.label.booktitle
775   emphasize
776 }

```

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

```

777 FUNCTION {format.in.ed.booktitle}
778 { format.booktitle
779   duplicate$ empty$
780   'skip$
781   {

```

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

```

782 is.online
783 { " " *
784   bbl.online *
785   #1 'online.shown :=
786   }
787   'skip$
788   if$

```

```

789 editor "editor" format.names.ed duplicate$ empty$ 'pop$
790 { " " *
791   " " *
792   get.bbl.editor add.period$

```

```

793     " " *
794     * swap$
795     *
796     word.in swap$ *
797   }
798   if$
799 }
800 if$
801 }

```

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

```

802 FUNCTION {legal.journal.shape}
803 { duplicate$ "OJ" =
804   'emphasize
805   'skip$
806   if$
807 }

```

7.7 Dates

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

```

765 FUNCTION {nodate.check}
766 {
767   skip$
768 }

```

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.

```

808 FUNCTION {nodate.check}
809 {
810   duplicate$ empty$
811   sortyear empty$
812   and
813   { options field.or.null "nonodate" =
814     'skip$
815     { pop$ bbl.nodate }
816     if$
817   }
818   'skip$
819   if$
820 }

```

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

```

821 FUNCTION {format.date}
822 { year "year" bibinfo.check nodate.check
823   extra.label *
824 }

```

`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.

```

825 FUNCTION {format.date.brackets}
826 { format.date
827   "[" swap$ * "]" *
828 }
829 FUNCTION {format.date.parens}
830 { format.date
831   "(" swap$ * ")" *
832 }
833 FUNCTION {format.jur.date.essential}
834 { options field.or.null "scottish-style" =

```



```

835 keywords field.or.null "sc" =
836 or
837   'format.date
838   'format.date.brackets
839   if$
840 }
841 FUNCTION {format.jur.date}
842 { journal field.or.null 's :=
843   s "OJ" =
844   s "ECR" =
845   or
846     'format.date.brackets
847     { options field.or.null "year-essential=false" =
848       'format.date.parens
849       { volume empty$
850         'format.jur.date.essential
851         { options field.or.null "year-essential=true" =
852           'format.jur.date.essential
853           'format.date.parens
854           if$
855         }
856       }
857     }
858   if$
859 }
860 if$
861 }

```

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

```

862 FUNCTION {format.eu.date}
863 { journal field.or.null 's :=
864   s "OJ" =
865   s "ECR" =
866   or
867     'format.date.brackets
868     'format.date
869   if$
870 }

```

7.8 Series, volumes, editions

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

```

871 FUNCTION {format.series}
872 { series
873   duplicate$ empty$ 'skip$
874   { "t" change.case$ }
875   if$
876   "series" bibinfo.check
877 }

```

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

```

878 FUNCTION {format.bvolume}
879 { volume empty$
880   { "" }
881   { bbl.volume volume tie.or.space.prefix
882     "volume" bibinfo.check * *
883     format.series
884     duplicate$ empty$ 'pop$
885     { emphasize ", " * swap$ * }
886     if$
887     "volume and number" number either.or.check
888   }
889   if$
890 }

```

`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.

```

891 FUNCTION {format.number.series}
892 { volume empty$
893   { number empty$
894     { series field.or.null }
895     { series empty$
896       { number "number" bibinfo.check }
897       { format.series
898         number tie.or.space.prefix "number" bibinfo.check * *
899       }
900     }
901   }
902   if$
903 }
904 { "" }
905 if$
906 }

```

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

```

907 FUNCTION {format.series.episode}
908 { series
909   duplicate$ empty$
910   'skip$
911   { "t" change.case$ }
912   if$
913   "series" bibinfo.check emphasize
914   number "number" bibinfo.check emphasize
915   duplicate$ empty$
916   { * }
917   { swap$
918     duplicate$ empty$
919     { * }
920     { ", " * swap$ * }
921   }
922   if$
923 }
924 }

```

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

```

925 FUNCTION {is.num}
926 { chr.to.int$
927   duplicate$ "0" chr.to.int$ < not
928   swap$ "9" chr.to.int$ > not and
929 }

```

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

```

930 FUNCTION {extract.num}
931 { duplicate$ 't :=
932   "" 's :=
933   { t empty$ not }
934   { t #1 #1 substring$
935     t #2 global.max$ substring$ 't :=
936     duplicate$ is.num
937     { s swap$ * 's := }
938     { pop$ "" 't := }
939   }
940   if$
941   while$
942   s empty$
943   'skip$
944   { pop$ s }
945   if$
946 }

```

`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.

```

947 FUNCTION {convert.edition}
948 { extract.num "1" change.case$ 's :=
949   s "first" = s "1" = or
950   { bbl.first 't := }
951   { s "second" = s "2" = or
952     { bbl.second 't := }
953     { s "third" = s "3" = or
954       { bbl.third 't := }
955       { s "fourth" = s "4" = or
956         { bbl.fourth 't := }
957         { s "fifth" = s "5" = or
958           { bbl.fifth 't := }
959           { s #1 #1 substring$ is.num
960             { s eng.ord 't := }
961             { edition 't := }
962             if$
963           }
964         if$
965       }
966     if$
967   }
968   if$
969 }
970 if$
971 }
972 if$
973 t
974 }

```

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

```

975 FUNCTION {format.edition}
976 { edition duplicate$ empty$ 'skip$
977   { convert.edition
978     output.state mid.sentence =
979     { "1" }
980     { "t" }
981     if$ change.case$
982     "edition" bibinfo.check
983     " " * bbl.edition *
984   }
985   if$
986 }

```

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.

```

987 FUNCTION {format.thesis.type}
988 { type duplicate$ empty$
989   'pop$
990   { swap$ pop$
991     "t" change.case$ "type" bibinfo.check
992   }
993   if$
994 }

```

`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 preceeding 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', 'Cmdnd' 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.

```

995 FUNCTION {add.cmd.number}
996 { " " swap$ "number" bibinfo.check *
997 }
998 FUNCTION {format.legr.number}
999 { series field.or.null 's :=
1000   s "C" =
1001     'add.cmd.number

```

```

1002 { s "Cd" =
1003     'add.cmd.number
1004     { s "Cmd" =
1005         'add.cmd.number
1006         { s "Cmnd" =
1007             'add.cmd.number
1008             { s "Cm" =
1009                 'add.cmd.number
1010                 { s purify$ #1 #3 substring$ 't :=
1011                     t "HL " =
1012                         { ", (" swap$
1013                             "number" bibinfo.check *
1014                             ")" *
1015                         }
1016                         { ", " swap$
1017                             "number" bibinfo.check *
1018                         }
1019                     if$
1020                 }
1021             if$
1022         }
1023     if$
1024 }
1025 if$
1026 }
1027 if$
1028 }
1029 if$
1030 }
1031 FUNCTION {format.tr.number}
1032 { series duplicate$ empty$
1033     { pop$
1034         type duplicate$ empty$
1035         { pop$
1036             number duplicate$ empty$
1037             { pop$ ""
1038             }
1039             { "(" swap$
1040                 "number" bibinfo.check
1041                 * ")" *
1042             }
1043             if$
1044         }
1045         { "(" swap$
1046             "t" change.case$ "type" bibinfo.check
1047             *
1048             number duplicate$ empty$
1049             'pop$
1050             { tie.or.space.prefix
1051                 "number" bibinfo.check
1052                 * *
1053             }
1054             if$
1055             ")" *
1056         }
1057         if$
1058     }
1059     { "t" change.case$ "(" swap$ *
1060         type duplicate$ empty$
1061         { pop$
1062             number duplicate$ empty$
1063             'pop$
1064             { format.legr.number * }
1065             if$
1066         }
1067         { "t" change.case$ "type" bibinfo.check
1068             *
1069             number duplicate$ empty$
1070             'pop$
1071             { tie.or.space.prefix
1072                 "number" bibinfo.check
1073                 * *
1074             }

```

```

1075         if$
1076     }
1077     if$
1078     ")" *
1079     continue.clause
1080 }
1081 if$
1082 }

```

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

```

1083 FUNCTION {format.manual.number}
1084 { series duplicate$ empty$
1085   { pop$
1086     type duplicate$ empty$
1087     { pop$
1088       number duplicate$ empty$
1089       { pop$ ""
1090       }
1091       { "number" bibinfo.check
1092       }
1093       if$
1094     }
1095     { "t" change.case$ "type" bibinfo.check
1096       number duplicate$ empty$
1097       'pop$
1098       { tie.or.space.prefix
1099         "number" bibinfo.check
1100         * *
1101       }
1102       if$
1103     }
1104     if$
1105   }
1106   { "t" change.case$
1107     number duplicate$ empty$
1108     { pop$
1109     }
1110     { tie.or.space.prefix
1111       "number" bibinfo.check
1112       * *
1113     }
1114     if$
1115     type duplicate$ empty$
1116     'pop$
1117     { ", " swap$ *
1118       "t" change.case$ "type" bibinfo.check
1119       *
1120     }
1121     if$
1122   }
1123   if$
1124 }

```

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

```

1125 FUNCTION {format.chapter}
1126 { chapter duplicate$ empty$
1127   { pop$ "" }
1128   { "chapter" bibinfo.check
1129     "c." swap$ *
1130   }
1131   if$
1132 }

```

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

```

1133 FUNCTION {format.series.number.chapter}
1134 { series duplicate$ empty$
1135   { pop$ "(" }

```

```

1136 { "series" bibinfo.check
1137   "(" swap$ *
1138 }
1139 if$
1140 type duplicate$ empty$
1141 { pop$
1142   number duplicate$ empty$
1143   'pop$
1144   { "number" bibinfo.check
1145     swap$ duplicate$ "(" =
1146     { swap$ * }
1147     { ", " * swap$ * }
1148     if$
1149   }
1150   if$
1151   chapter duplicate$ empty$
1152   'pop$
1153   { "chapter" bibinfo.check
1154     swap$ duplicate$ "(" =
1155     { swap$ * }
1156     { ", c." * swap$ * }
1157     if$
1158   }
1159   if$
1160 }
1161 { "type" bibinfo.check
1162   swap$ duplicate$ "(" =
1163   { swap$ * }
1164   { ", " * swap$ * }
1165   if$
1166   number duplicate$ empty$
1167   { pop$
1168     chapter duplicate$ empty$
1169     'pop$
1170     { "chapter" bibinfo.check
1171       " c." swap$ * *
1172     }
1173     if$
1174   }
1175   { "number" bibinfo.check
1176     " " swap$ * *
1177     chapter duplicate$ empty$
1178     'pop$
1179     { "chapter" bibinfo.check
1180       ", c." swap$ * *
1181     }
1182     if$
1183   }
1184   if$
1185 }
1186 if$
1187 ")" *
1188 duplicate$ "(" =
1189 { pop$ "" }
1190 'skip$
1191 if$
1192 }

```

`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'

```

1193 FUNCTION {format.case.number}
1194 { casenumber duplicate$ empty$
1195   { pop$
1196     number duplicate$ empty$
1197     { pop$ "" }
1198     { institution field.or.null "Commission" =
1199       userb empty$ not
1200       and
1201       { userb "userb" bibinfo.check
1202         "(" swap$ * " " * bbl.commission * " " *
1203         swap$ "number" bibinfo.check *
1204       }

```

```

1205         { "number" bibinfo.check
1206           "(" swap$ * ")" *
1207         }
1208         if$
1209       }
1210     if$
1211   }
1212   { "casenumber" bibinfo.check
1213     "(" swap$ * ")" *
1214     institution field.or.null "Commission" =
1215     number empty$ not
1216     and
1217     { " " * bbl.commission * " " *
1218       number "number" bibinfo.check *
1219     }
1220     'skip$
1221     if$
1222   }
1223   if$
1224 }

```

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

```

1225 FUNCTION {eu.case.check}
1226 { journal field.or.null "OJ" =
1227   number empty$ not
1228   or
1229   casenumber empty$ not
1230   or
1231   'continue.clause
1232   'new.block
1233   if$
1234 }

```

7.9 Notes

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

```

1235 FUNCTION {format.note}
1236 { note empty$
1237   { "" }
1238   { note #1 #1 substring$
1239     duplicate$ "{" =
1240     'skip$
1241     { output.state mid.sentence =
1242       output.state mid.clause =
1243       or
1244       { "l" }
1245       { "u" }
1246       if$
1247       change.case$
1248     }
1249     if$
1250     note #2 global.max$ substring$ * "note" bibinfo.check
1251   }
1252   if$
1253 }

```

7.10 Addresses and publishers

format.org.or.pub intercepts a publisher (or whatever) being written out as bibinfo. If the write will succeed and an address has been specified, the address is written out first, then a colon and space; then the original write is allowed to proceed.

```

1254 FUNCTION {format.org.or.pub}
1255 { 't :=
1256   ""

```

```

1257 address empty$ t empty$ and
1258 'skip$
1259 { address "address" bibinfo.check *
1260   t empty$
1261   'skip$
1262   { address empty$
1263     'skip$
1264     { ": " * }
1265     if$
1266     t *
1267   }
1268   if$
1269 }
1270 if$
1271 }

```

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

```

1272 FUNCTION {format.publisher.address}
1273 { publisher "publisher" bibinfo.warn format.org.or.pub
1274 }
1275 FUNCTION {format.organization.address}
1276 { publisher empty$
1277   { organization "organization" bibinfo.check format.org.or.pub }
1278   { publisher "publisher" bibinfo.warn format.org.or.pub
1279     organization duplicate$ empty$
1280     'pop$
1281     { "organization" bibinfo.check ". " * swap$ * }
1282     if$
1283   }
1284   if$
1285 }
1286 FUNCTION {format.institution.address}
1287 { publisher empty$
1288   { institution "institution" bibinfo.check format.org.or.pub }
1289   { publisher "publisher" bibinfo.warn format.org.or.pub
1290     institution duplicate$ empty$
1291     'pop$
1292     { "institution" bibinfo.check ". " * swap$ * }
1293     if$
1294   }
1295   if$
1296 }

```

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

```

1297 FUNCTION {format.library}
1298 { library "library" bibinfo.check
1299   emphasize

```

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

```

1300 duplicate$ empty$ not
1301 is.online
1302 and
1303 journal empty$
1304 and
1305 { " " * bbl.online *
1306   #1 'online.shown :=
1307 }
1308 'skip$
1309 if$

```

The library 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.

```

1310 }
1311 FUNCTION {format.img.library}
1312 { library duplicate$ empty$

```



```

1313     { pop$ "" }
1314     { "library" bibinfo.check
1315       "At: " swap$ *
1316     }
1317   if$
1318 }
1319 FUNCTION {format.img.library.address}
1320 { address empty$ library empty$ and institution empty$ and
1321   { "" }
1322   { "At: "
1323     address "address" bibinfo.check *
1324     library duplicate$ empty$
1325       { institution duplicate$ empty$
1326         { pop$ "" }
1327         { address empty$
1328           'skip$
1329           { ". " swap$ }
1330           if$
1331           "institution" bibinfo.check * *
1332         }
1333       }
1334     }
1335     { address empty$
1336       'skip$
1337       { ". " swap$ }
1338       if$
1339       "library" bibinfo.check * *
1340     }
1341   }
1342 }
1343 if$
1344 }
1345 FUNCTION {format.pub.org.lib.address}
1346 { publisher duplicate$ empty$
1347   { pop$
1348     organization duplicate$ empty$
1349     { pop$
1350       format.img.library.address }
1351     { "organization" bibinfo.check
1352       library empty$
1353       { format.org.or.pub }
1354       { ". " *
1355         format.img.library.address *
1356       }
1357     }
1358   }
1359   if$
1360 }
1361 { "publisher" bibinfo.warn format.org.or.pub
1362   library empty$
1363   'skip$
1364   { ". " *
1365     format.img.library *
1366   }
1367   if$
1368 }
1369 if$
1370 }

```

7.11 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.

```

1372 INTEGERS { multiresult }
1373 FUNCTION {multi.page.check}
1374 { 't :=
1375   #0 'multiresult :=
1376   { multiresult not
1377     t empty$ not
1378     and

```

```

1379     }
1380     { t #1 #1 substring$
1381       duplicate$ "-" =
1382       swap$ duplicate$ ", " =
1383       swap$ "+" =
1384       or or
1385       { #1 'multiresult := }
1386       { t #2 global.max$ substring$ 't := }
1387       if$
1388     }
1389     while$
1390     multiresult
1391   }

```

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

```

1392 FUNCTION {n.dashify}
1393 { 't :=
1394   ""
1395   { t empty$ not }
1396   { t #1 #1 substring$ "-" =
1397     { t #1 #2 substring$ "--" = not
1398       { "--" *
1399         t #2 global.max$ substring$ 't :=
1400       }
1401       { { t #1 #1 substring$ "-" = }
1402         { "-" *
1403           t #2 global.max$ substring$ 't :=
1404         }
1405       while$
1406     }
1407     if$
1408   }
1409   { t #1 #1 substring$ *
1410     t #2 global.max$ substring$ 't :=
1411   }
1412   if$
1413 }
1414 while$
1415 }

```

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

```

1416 FUNCTION {format.pages}
1417 { pages duplicate$ empty$ 'skip$
1418   { duplicate$ multi.page.check
1419     { bbl.pages swap$
1420       n.dashify
1421     }
1422     { bbl.page swap$
1423     }
1424     if$
1425     "pages" bibinfo.check
1426     *
1427   }
1428   if$
1429 }

```

`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.

```

1430 FUNCTION {format.journal.pages}
1431 { pages duplicate$ empty$ 'pop$
1432   { swap$ duplicate$ empty$
1433     { pop$ pop$ format.pages }
1434     { ", " *
1435       swap$
1436       n.dashify
1437       pages multi.page.check
1438       'bbl.pages
1439       'bbl.page
1440       if$
1441       swap$

```

```

1442         "pages" bibinfo.check
1443     * *
1444 }
1445 if$
1446 }
1447 if$
1448 }

```

`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).

```

1449 FUNCTION {format.journal.eid}
1450 { eid "eid" bibinfo.check
1451   duplicate$ empty$ 'pop$
1452   { swap$ duplicate$ empty$ 'skip$
1453     { ", " *
1454     }
1455     if$
1456     swap$ *
1457   }
1458   if$
1459 }

```

`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.

```

1460 FUNCTION {format.vol.num.pages}
1461 { pubstate field.or.null
1462   duplicate$ "inpress" =
1463   { pop$ bbl.inpress }
1464   { duplicate$ "submitted" =
1465     { pop$ bbl.submitted }
1466     { duplicate$ "inpreparation" =
1467       { pop$ bbl.inpreparation }
1468       { pop$ "" }
1469     if$
1470     }
1471   if$
1472 }
1473 if$
1474 duplicate$ empty$
1475 { pop$
1476   volume field.or.null
1477   duplicate$ empty$ 'skip$
1478   { "volume" bibinfo.check
1479   }
1480   if$
1481   number "number" bibinfo.check duplicate$ empty$ 'skip$
1482   { swap$ duplicate$ empty$
1483     { "there's a number but no volume in " cite$ * warning$ }
1484     'skip$
1485     if$
1486     swap$
1487     "(" swap$ * ")" *
1488   }
1489   if$ *
1490   month "month" bibinfo.check duplicate$ empty$
1491   'pop$
1492   { swap$ duplicate$ empty$
1493     'pop$
1494     { ", " * swap$ * }
1495     if$
1496   }
1497   if$
1498   eid empty$
1499   { format.journal.pages }

```

```

1500     { format.journal.eid }
1501     if$
1502   }
1503   'skip$
1504   if$
1505 }

```

`format.chapter.pages` returns pinpointing information for books, in the form ‘chap. chapter, p.pages’.

```

1506 FUNCTION {format.chapter.pages}
1507 { chapter empty$
1508   'format.pages
1509   { type empty$
1510     { bbl.chapter }
1511     { type "1" change.case$
1512       "type" bibinfo.check
1513     }
1514     if$
1515     chapter tie.or.space.prefix
1516     "chapter" bibinfo.check
1517     * *
1518     pages empty$
1519     'skip$
1520     { ", " * format.pages * }
1521     if$
1522   }
1523   if$
1524 }

```

`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.

```

1525 FUNCTION {format.journal.series.vol.pages}
1526 { journal field.or.null
1527   duplicate$ "OJ" =
1528   { "journal" bibinfo.check
1529     emphasize
1530     series empty$
1531     volume empty$
1532     and
1533     pages empty$
1534     and
1535     'skip$
1536     { " " * }
1537     if$
1538     format.series
1539     volume "volume" bibinfo.check *
1540     duplicate$ empty$
1541     'skip$
1542     { pages field.or.null duplicate$ empty$
1543       'skip$
1544       { "/" swap$ *
1545         }
1546       if$ *
1547     }
1548     if$ *
1549   }
1550   { duplicate$ "ECR" =
1551     { "journal" bibinfo.check
1552       volume "volume" bibinfo.check
1553       duplicate$ empty$
1554       'skip$
1555       { " " swap$ * "--" * * }
1556       if$
1557     }
1558     { "journal" bibinfo.check

```

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

```

1559     is.online
1560     { " " * bbl.online *
1561       #1 'online.shown :=
1562     }
1563     'skip$
1564     if$

1565     volume "volume" bibinfo.check
1566     duplicate$ empty$
1567     'pop$
1568     { " " * swap$ * }
1569     if$
1570     pages empty$
1571     eid empty$
1572     and
1573     'skip$
1574     { " " * }
1575     if$
1576   }
1577   if$
1578 }
1579 if$
1580 eid empty$
1581 { pages "pages" bibinfo.check * }
1582 { eid "eid" bibinfo.check * }
1583 if$
1584 }

```

7.12 URLs

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

```

1585 FUNCTION {format.eprint}
1586 { eprint duplicate$ empty$
1587   'skip$
1588   { "\eprint"
1589     archive empty$
1590     'skip$
1591     { "[" * archive * "]" * }
1592     if$
1593     "{" * swap$ * "}" *
1594   }
1595   if$
1596 }

```

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

```

1597 FUNCTION {format.url}
1598 { doi empty$
1599   { url }
1600   { "https://doi.org/" doi * }
1601   if$
1602   duplicate$ empty$
1603   { pop$ "" }
1604   { "\urlprefix\url{" swap$ * "}" * }
1605   if$
1606   duplicate$ empty$
1607   urlyear empty$ not
1608   urldate empty$ not
1609   or
1610   and
1611   { library duplicate$ empty$
1612     'pop$
1613     { "\urlprefix{" swap$ * * }
1614     if$
1615   }
1616   'skip$
1617   if$
1618   urlyear empty$
1619   { urldate empty$

```

```

1620     'skip$
1621     { " [\urldateprefix{}]" * urldate * "]" * }
1622   if$
1623 }
1624 { " [\urldateprefix{}]" * urlyear * "]" * }
1625 if$
1626 }

```

7.13 Cross referencing between entries

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).

```

1627 FUNCTION {format.article.crossref}
1628 { word.in
1629   " \cite{" * crossref * "}" *
1630 }
1631 FUNCTION {format.incoll.inproc.crossref}
1632 { word.in
1633   " \cite{" * crossref * "}" *
1634 }

```

`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.

```

1635 FUNCTION {format.book.crossref}
1636 { volume duplicate$ empty$
1637   { "empty volume in " cite$ * "'s crossref of " * crossref * warning$
1638     pop$ word.in
1639   }
1640   { bbl.volume
1641     capitalize
1642     swap$ tie.or.space.prefix "volume" bibinfo.check * * bbl.of space.word *
1643   }
1644   if$
1645   " \cite{" * crossref * "}" *
1646 }

```

7.14 Formatting drivers for specific entry types

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

7.14.1 Article

```

1648 FUNCTION {article}
1649 { output.bibitem
1650   format.authors "author" output.check
1651   author format.key output
1652   format.date "year" output.check
1653   date.block
1654   format.title "title" output.check
1655   get.title.addenda
1656   new.block
1657   crossref missing$
1658     { journal "t" change.case$
1659       "journal" bibinfo.check
1660       emphasize
1661       "journal" output.check

```

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

```

1662   is.online
1663     { continue.clause
1664       bbl.online output
1665       #1 'online.shown :=
1666     }
1667   'skip$
1668   if$

```

```

1669     format.vol.num.pages output
1670   }
1671   { format.article.crossref output.nonnull
1672     format.pages output
1673   }
1674   if$
1675   new.block
1676   format.note output
1677   new.block
1678   format.eprint output
1679   format.url output
1680   fin.entry
1681 }

```

7.14.2 Book

```

1682 FUNCTION {book}
1683 { output.bibitem
1684   author empty$
1685   { editor empty$
1686     type$ "reference" =
1687     or
1688     { type$ "reference" =
1689       { format.title "title" output.check }
1690       { format.btitle "title" output.check }
1691       if$
1692       get.title.online
1693       format.date "year" output.check
1694       year empty$ 'skip$ { date.block } if$
1695       get.title.addenda
1696     }
1697     { format.editors "author and editor" output.check
1698       format.date "year" output.check
1699       date.block
1700       format.btitle "title" output.check
1701       get.title.addenda
1702     }
1703     if$
1704   }
1705   { format.authors output.nonnull
1706     crossref missing$
1707     { "author and editor" editor either.or.check }
1708     'skip$
1709     if$
1710     format.date "year" output.check
1711     date.block
1712     format.btitle "title" output.check
1713     get.title.addenda
1714   }
1715   if$
1716   format.bvolume output
1717   format.number.series output
1718   new.block
1719   format.edition output
1720   crossref missing$
1721   { new.sentence
1722     format.publisher.address output
1723   }
1724   { new.block
1725     format.book.crossref output.nonnull
1726   }
1727   if$
1728   new.block
1729   format.note output
1730   new.block
1731   format.eprint output
1732   format.url output
1733   fin.entry
1734 }

```

7.14.3 Unpublished or self-published monograph

```
1735 FUNCTION {booklet}
1736 { output.bibitem
1737   author empty$
1738   { format.avtitle "title" output.check
1739     get.title.online
1740     format.date "year" output.check
1741     year empty$ 'skip$ { date.block } if$
1742     get.title.addenda
1743   }
1744   { format.authors output
1745     author format.key output
1746     format.date "year" output.check
1747     date.block
1748     format.avtitle "title" output.check
1749     get.title.addenda
1750   }
1751   if$
1752   new.block
1753   format.series.episode output
```

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

```
1754 is.online
1755   { continue.clause
1756     bbl.online output
1757     #1 'online.shown :=
1758   }
1759   'skip$
1760   if$
```

```
1761 new.block
1762 type "type" bibinfo.check output
1763 new.block
1764 howpublished "howpublished" bibinfo.check output
1765 new.block
1766 format.note output
1767 publisher empty$ 'skip$
1768   { new.block}
1769   if$
1770 format.publisher.address output
1771 new.block
1772 format.eprint output
1773 format.url output
1774 fin.entry
1775 }
```

7.14.4 Work in book

```
1776 FUNCTION {inbook}
1777 { output.bibitem
1778   author empty$
1779   { format.editors "author and editor" output.check
1780     editor format.key output
1781   }
1782   { format.authors output.nonnull
1783     crossref missing$
1784     { "author and editor" editor either.or.check }
1785     'skip$
1786     if$
1787   }
1788   if$
1789   format.date "year" output.check
1790   date.block
1791   format.btitle "title" output.check
1792   get.title.addenda
1793   new.block
1794   format.edition output
1795   crossref missing$
1796   { format.publisher.address output
```



```

1797     format.bvolume output
1798     format.chapter.pages "chapter and pages" output.check
1799     new.block
1800     format.number.series output
1801     new.sentence
1802   }
1803   { format.chapter.pages "chapter and pages" output.check
1804     new.block
1805     format.book.crossref output.nonnull
1806   }
1807   if$
1808   new.block
1809   format.note output
1810   new.block
1811   format.eprint output
1812   format.url output
1813   fin.entry
1814 }

```

7.14.5 Work in collection

```

1815 FUNCTION {incollection}
1816 { output.bibitem
1817   author empty$
1818   { type$ "inreference" =
1819     { format.label.booktitle output}
1820     { format.title "title" output.check
1821       get.title.online
1822     }
1823     if$
1824     format.date "year" output.check
1825     year empty$ 'skip$ { date.block } if$
1826     type$ "inreference" =
1827     'skip$
1828     { get.title.addenda }
1829     if$
1830   }
1831   { format.authors output.nonnull
1832     format.date "year" output.check
1833     date.block
1834     format.title "title" output.check
1835     get.title.addenda
1836   }
1837   if$
1838   new.block
1839   crossref missing$
1840   { type$ "inreference" =
1841     author empty$
1842     and
1843     { format.byeditors output }
1844     { format.in.ed.booktitle "booktitle" output.check }
1845     if$
1846     new.block
1847     format.edition output
1848     new.block
1849     type$ "inreference" =
1850     author empty$
1851     and
1852     { format.btitle "title" output.check
1853       get.title.addenda
1854       new.block
1855     }
1856     'skip$
1857     if$
1858     format.publisher.address output
1859     format.bvolume output
1860     format.number.series output
1861     format.chapter.pages output
1862   }
1863   { format.incoll.inproc.crossref output.nonnull
1864     format.chapter.pages output

```

```

1865     }
1866     if$
1867     new.block
1868     format.note output
1869     new.block
1870     format.eprint output
1871     format.url output
1872     fin.entry
1873 }

```

7.14.6 Conference paper

```

1874 FUNCTION {inproceedings}
1875 { output.bibitem
1876   format.authors "author" output.check
1877   author format.key output
1878   format.date "year" output.check
1879   date.block
1880   format.title "title" output.check
1881   get.title.addenda
1882   new.block
1883   crossref missing$
1884     { format.in.ed.booktitle "booktitle" output.check
1885       eventyear output
1886       venue output
1887       new.sentence
1888       publisher empty$
1889         { format.organization.address output }
1890         { organization "organization" bibinfo.check output
1891           format.publisher.address output
1892         }
1893       if$
1894       format.bvolume output
1895       format.number.series output
1896       format.pages output
1897     }
1898     { format.incoll.inproc.crossref output.nonnull
1899       format.pages output
1900     }
1901   if$
1902   new.block
1903   format.note output
1904   new.block
1905   format.eprint output
1906   format.url output
1907   fin.entry
1908 }
1909 FUNCTION {conference} { inproceedings }

```

7.14.7 Legal case studies

```

1910 FUNCTION {jurisdiction}
1911 { output.bibitem
1912   author empty$
1913     { format.btitle "title" output.check
1914       continue.clause
1915       format.case.number output
1916       number empty$
1917       casenumber empty$
1918       and
1919       'new.sentence
1920       'continue.clause
1921     if$
1922     get.title.online
1923     format.jur.date "year" output.check
1924     year empty$ 'skip$ { eu.case.check } if$
1925     get.title.addenda
1926   }
1927   { format.authors output

```

```

1928     author format.key output
1929     format.jur.date "year" output.check
1930     date.block
1931     continue.clause
1932     format.btitle "title" output.check
1933     get.title.addenda
1934   }
1935   if$
1936   eu.case.check
1937   format.note output
1938   note empty$
1939   'eu.case.check
1940   'new.block
1941   if$
1942   crossref missing$
1943     { continue.clause
1944       format.journal.series.vol.pages output
1945     }
1946     { format.article.crossref output.nonnull
1947       format.pages output
1948     }
1949   if$
1950   new.block
1951   format.eprint output
1952   format.url output
1953   fin.entry
1954 }

```

7.14.8 Legislation

```

1955 FUNCTION {uklegislation}
1956 { output.bibitem
1957   author empty$
1958   { format.btitle "title" output.check
1959     continue.clause
1960     format.date emphasize "year" output.check
1961     get.title.addenda
1962   }
1963   { format.authors output
1964     author format.key output
1965     format.date "year" output.check
1966     date.block
1967     continue.clause
1968     format.btitle "title" output.check
1969     get.title.addenda
1970   }
1971   if$
1972   entrysubtype field.or.null "secondary" =
1973   { number "number" bibinfo.check output }
1974   { series empty$ type empty$ and
1975     { number empty$
1976       { format.chapter output }
1977       { continue.clause
1978         format.series.number.chapter output
1979       }
1980     }
1981   }
1982   { chapter empty$
1983     { new.block }
1984     { continue.clause }
1985   }
1986   if$
1987   format.series.number.chapter output
1988 }
1989 if$
1990 if$
1991 note empty$
1992 'skip$
1993 { new.block
1994   format.note output
1995   new.block

```

```

1996     }
1997     if$
1998     publisher empty$ 'skip$
1999     { entrysubtype field.or.null "secondary" =
2000       'skip$
2001       { new.block }
2002       if$
2003       format.publisher.address output
2004     }
2005     if$
2006     new.block
2007     format.eprint output
2008     format.url output
2009     fin.entry
2010   }
2011   FUNCTION {eulegislation}
2012   { output.bibitem
2013     author empty$
2014     { format.title "title" output.check
2015       continue.clause
2016       format.eu.date "year" output.check
2017       get.title.addenda
2018     }
2019     { format.authors output
2020       author format.key output
2021       format.eu.date "year" output.check
2022       date.block
2023       continue.clause
2024       format.title "title" output.check
2025       get.title.addenda
2026     }
2027     if$
2028     eu.case.check
2029     format.note output
2030     note empty$
2031     'eu.case.check
2032     'new.block
2033     if$
2034     crossref missing$
2035     { continue.clause
2036       format.journal.series.vol.pages output
2037     }
2038     { format.article.crossref output.nonnull
2039       format.pages output
2040     }
2041     if$
2042     new.block
2043     format.eprint output
2044     format.url output
2045     fin.entry
2046   }
2047   FUNCTION {legislation}
2048   { journal empty$
2049     'uklegislation
2050     'eulegislation
2051     if$
2052   }

```

7.14.9 Manual

```

2053   FUNCTION {manual}
2054   { output.bibitem
2055     author empty$
2056     { format.btitle "title" output.check
2057       get.title.online
2058       format.date "year" output.check
2059       year empty$ 'skip$ { date.block } if$
2060       get.title.addenda
2061     }
2062     { format.authors output
2063       author format.key output

```

```

2064     format.date "year" output.check
2065     date.block
2066     format.btitle "title" output.check
2067     get.title.addenda
2068   }
2069   if$
2070   month "month" bibinfo.check output
2071   format.edition output
2072   new.block
2073   format.manual.number output
2074   new.block
2075   format.library output
2076   organization address new.block.checkb
2077   format.organization.address output
2078   new.block
2079   format.note output
2080   new.block
2081   format.eprint output
2082   format.url output
2083   fin.entry
2084 }

```

7.14.10 Image

```

2085 FUNCTION {image}
2086 { output.bibitem
2087   author empty$
2088   { format.btitle "title" output.check
2089     get.title.online
2090     format.date "year" output.check
2091     year empty$ 'skip$ { date.block } if$
2092     get.title.addenda
2093   }
2094   { format.authors output
2095     author format.key output
2096     format.date "year" output.check
2097     date.block
2098     format.btitle "title" output.check
2099     get.title.addenda
2100   }
2101   if$
2102   format.edition output
2103   new.block
2104   format.manual.number output
2105   new.block
2106   format.pub.org.lib.address output
2107   new.block
2108   format.note output
2109   new.block
2110   format.eprint output
2111   format.url output
2112   fin.entry
2113 }

```

7.14.11 Master's thesis

```

2114 FUNCTION {mastersthesis}
2115 { output.bibitem
2116   format.authors "author" output.check
2117   author format.key output
2118   format.date "year" output.check
2119   date.block
2120   format.btitle "title" output.check
2121   get.title.addenda
2122   new.block
2123   bbl.mthesis format.thesis.type output.nonnull
2124   new.block
2125   school "school" bibinfo.warn output
2126   address "address" bibinfo.check output
2127   new.block

```

```

2128     format.note output
2129     new.block
2130     format.eprint output
2131     format.url output
2132     fin.entry
2133 }

```

7.14.12 Miscellaneous

```

2134 FUNCTION {misc}
2135 { output.bibitem
2136   author empty$
2137   { format.title "title" output.check
2138     get.title.online
2139     format.date "year" output.check
2140     year empty$ 'skip$ { date.block } if$
2141     get.title.addenda
2142   }
2143   { format.authors output
2144     author format.key output
2145     format.date "year" output.check
2146     date.block
2147     format.title "title" output.check
2148     get.title.addenda
2149   }
2150   if$
2151   new.block
2152   howpublished "howpublished" bibinfo.check output
2153   new.block
2154   format.note output
2155   new.block
2156   format.eprint output
2157   format.url output
2158   fin.entry
2159 }

```

7.14.13 Doctoral thesis

```

2160 FUNCTION {phdthesis}
2161 { output.bibitem
2162   format.authors "author" output.check
2163   author format.key output
2164   format.date "year" output.check
2165   date.block
2166   format.btitle "title" output.check
2167   get.title.addenda
2168   new.block
2169   bbl.phdthesis format.thesis.type output.nonnull
2170   new.block
2171   school "school" bibinfo.warn output
2172   address "address" bibinfo.check output
2173   new.block
2174   format.note output
2175   new.block
2176   format.eprint output
2177   format.url output
2178   fin.entry
2179 }

```

7.14.14 Conference proceedings

```

2180 FUNCTION {proceedings}
2181 { output.bibitem
2182   editor empty$
2183   { format.btitle "title" output.check
2184     get.title.online
2185     format.date "year" output.check
2186     year empty$ 'skip$ { date.block } if$

```

```

2187     get.title.addenda
2188 }
2189 { format.editors output
2190   editor format.key output
2191   format.date "year" output.check
2192   date.block
2193   format.btitle "title" output.check
2194   get.title.addenda
2195 }
2196 if$
2197 format.bvolume output
2198 format.number.series output
2199 new.sentence
2200 publisher empty$
2201 { format.organization.address output }
2202 { organization "organization" bibinfo.check output
2203   format.publisher.address output
2204 }
2205 if$
2206 new.block
2207 format.note output
2208 new.block
2209 format.eprint output
2210 format.url output
2211 fin.entry
2212 }

```

7.14.15 Report

```

2213 FUNCTION {techreport}
2214 { output.bibitem
2215   author empty$
2216   { format.btitle "title" output.check
2217     get.title.online
2218     format.date "year" output.check
2219     year empty$ 'skip$ { date.block } if$
2220     get.title.addenda
2221   }
2222   { format.authors output
2223     author format.key output
2224     format.date "year" output.check
2225     date.block
2226     format.btitle "title" output.check
2227     get.title.addenda
2228   }
2229   if$
2230   new.block
2231   format.tr.number output
2232   new.block
2233   publisher empty$
2234   { format.institution.address output }
2235   { institution "institution" bibinfo.check output
2236     format.publisher.address output
2237   }
2238   if$
2239   new.block
2240   format.library output
2241   new.block
2242   format.note output
2243   new.block
2244   format.eprint output
2245   format.url output
2246   fin.entry
2247 }

```

7.14.16 Unpublished works

```
2248 FUNCTION {unpublished}
2249 { output.bibitem
2250   author empty$
2251   { booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2252     get.title.online
2253     format.date "year" output.check
2254     year empty$ 'skip$ { date.block } if$
2255     get.title.addenda
2256   }
2257   { format.authors output
2258     author format.key output
2259     format.date "year" output.check
2260     date.block
2261     booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2262     get.title.addenda
2263   }
2264   if$
2265   new.block
2266   format.in.ed.booktitle output
2267   new.block
2268   howpublished "howpublished" bibinfo.check output
2269   new.block
2270   bbl.unpublished output
2271   new.block
2272   format.note output
2273   new.block
2274   format.eprint output
2275   format.url output
2276   fin.entry
2277 }
```

7.14.17 Aliases and fallbacks

Here are a list of aliases supported by this style.

```
2279 FUNCTION {letter}
2280 { journal empty$
2281   'unpublished
2282   'article
2283   if$
2284 }
2285 FUNCTION {collection} { book }
2286 FUNCTION {reference} { book }
2287 FUNCTION {inreference} { incollection }
2288 FUNCTION {audio} { booklet }
2289 FUNCTION {movie} { booklet }
2290 FUNCTION {music} { booklet }
2291 FUNCTION {video} { booklet }
2292 FUNCTION {dataset} { manual }
2293 FUNCTION {electronic} { manual }
2294 FUNCTION {online} { manual }
2295 FUNCTION {patent} { manual }
2296 FUNCTION {software} { manual }
2297 FUNCTION {standard} { manual }
2298 FUNCTION {www} { manual }
2299 FUNCTION {thesis} { phdthesis }
2300 FUNCTION {report} { techreport }
```

The fallback driver is misc, as with most styles.

```
2301 FUNCTION {default.type} { misc }
```

7.15 Bibliography processing

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

```
2303 READ
```


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

```
2304 FUNCTION {sortify}
2305 { purify$
2306   "l" change.case$
2307 }
```

`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*.

```
2308 INTEGERS { len }
2309 FUNCTION {chop.word}
2310 { 's :=
2311   'len :=
2312   s #1 len substring$ =
2313     { s len #1 + global.max$ substring$ }
2314     's
2315   if$
2316 }
```

`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.’

```
2317 FUNCTION {format.lab.names}
2318 { 's :=
2319   "" 't :=
2320   #1 'nameptr :=
2321   s num.names$ 'numnames :=
2322   numnames 'namesleft :=
2323   { namesleft #0 > }
2324   { s nameptr
2325     "{vv-}{ll}" format.name$
2326     't :=
2327     nameptr #1 >
2328     { nameptr #2 =
2329       numnames #3 > and
2330       { "others" 't :=
2331         #1 'namesleft := }
2332       'skip$
2333       if$
2334       namesleft #1 >
2335       { ", " * t * }
2336       { s nameptr "{ll}" format.name$ duplicate$ "others" =
2337         { 't := }
2338         { pop$ }
2339         if$
2340         t "others" =
2341         { " " * bbl.etal *
2342         }
2343         { bbl.and
2344           space.word * t *
2345         }
2346         if$
2347       }
2348       if$
2349     }
2350     't
2351     if$
2352     nameptr #1 + 'nameptr :=
2353     namesleft #1 - 'namesleft :=
2354   }
2355   while$
2356 }
```

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

```
2357 FUNCTION {author.editor.key.label}
2358 { author empty$
2359   { editor empty$
2360     { key empty$
2361       { title empty$
```

```

2362         { cite$ #1 #3 substring$ }
2363     { title
2364         type$ "book" =
2365         { emphasize }
2366         'skip$
2367         if$
2368     }
2369     if$
2370 }
2371 'key
2372 if$
2373 }
2374 { editor format.lab.names }
2375 if$
2376 }
2377 { author format.lab.names }
2378 if$
2379 }

```

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

```

2380 FUNCTION {editor.key.label}
2381 { editor empty$
2382   { key empty$
2383     { title empty$
2384       { cite$ #1 #3 substring$ }
2385       { title emphasize }
2386       if$
2387     }
2388     'key
2389     if$
2390   }
2391   { editor format.lab.names }
2392   if$
2393 }

```

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

```

2394 FUNCTION {author.key.label}
2395 { author empty$
2396   { key empty$
2397     { type$ "inreference" =
2398       booktitle empty$ not
2399       and
2400       { booktitle }
2401       { title empty$
2402         { cite$ #1 #3 substring$ }
2403         { title
2404           type$ "article" =
2405           type$ "incollection" =
2406           or
2407           type$ "inproceedings" =
2408           or
2409           type$ "reference" =
2410           author empty$
2411           and
2412           or
2413           type$ "misc" =
2414           or
2415           'skip$
2416           { emphasize }
2417           if$
2418         }
2419         if$
2420       }
2421       if$
2422     }
2423     'key
2424     if$
2425   }
2426   { author format.lab.names }

```

```
2427 if$
2428 }
```

calc.short.authors chooses between the above.

```
2429 FUNCTION {calc.short.authors}
2430 { type$ "book" =
2431   type$ "inbook" =
2432   or
2433     'author.editor.key.label
2434     { type$ "proceedings" =
2435       'editor.key.label
2436       'author.key.label
2437       if$
2438     }
2439   if$
2440   'short.list :=
2441 }
```

calc.label creates a string in the form ‘Truncated Author List(Year’. The closing parenthesis is supplied by output.bibitem (see above), into which this string is incorporated.

```
2442 FUNCTION {calc.label}
2443 { calc.short.authors
2444   short.list
2445   "("
2446   *
2447   year nodate.check duplicate$ empty$
2448     { pop$ "" }
2449     'skip$
2450   if$
2451   *
2452   'label :=
2453 }
```

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.’

```
2454 FUNCTION {sort.format.names}
2455 { 's :=
2456   #1 'nameptr :=
2457   ""
2458   s num.names$ 'numnames :=
2459   numnames 'namesleft :=
2460     { namesleft #0 > }
2461     { s nameptr
2462       "{ll{ }}{ f{ }}{ jj{ }}{ vv{}}}"
2463       format.name$ 't :=
2464       nameptr #1 >
2465       { " " *
2466         namesleft #1 =
2467         t "others" =
2468         and
2469         { "zzzzz" 't := }
2470         'skip$
2471       if$
2472       t sortify *
2473     }
2474     { t sortify * }
2475   if$
2476   nameptr #1 + 'nameptr :=
2477   namesleft #1 - 'namesleft :=
2478 }
2479 while$
2480 }
```

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

```
2481 FUNCTION {sort.format.title}
2482 { 't :=
2483   "A " #2
```

```

2484 "An " #3
2485 "The " #4 t chop.word
2486 chop.word
2487 chop.word
2488 sortify
2489 #1 global.max$ substring$
2490 }

```

`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.

```

2491 FUNCTION {author.editor.sort}
2492 { author empty$
2493   { editor empty$
2494     { title empty$
2495       { key empty$
2496         { "to sort, need author, editor, title, or key in " cite$ * warning$
2497           ""
2498         }
2499         { key sortify }
2500         if$
2501       }
2502       { title sort.format.title }
2503       if$
2504     }
2505     { editor sort.format.names }
2506     if$
2507   }
2508   { author sort.format.names }
2509   if$
2510 }

```

`editor.sort` does similar but skips the author names.

```

2511 FUNCTION {editor.sort}
2512 { editor empty$
2513   { title empty$
2514     { key empty$
2515       { "to sort, need editor, title, or key in " cite$ * warning$
2516         ""
2517       }
2518       { key sortify }
2519       if$
2520     }
2521     { title sort.format.title }
2522     if$
2523   }
2524   { editor sort.format.names }
2525   if$
2526 }

```

`author.sort` does similar but skips the editor names.

```

2527 FUNCTION {author.sort}
2528 { author empty$
2529   { type$ "inreference" =
2530     booktitle empty$ not
2531     and
2532     { booktitle sort.format.title }
2533     { title empty$
2534       { key empty$
2535         { "to sort, need author, title, or key in " cite$ * warning$
2536           ""
2537         }
2538         { key sortify }
2539         if$
2540       }
2541       { title sort.format.title }
2542       if$
2543     }
2544     if$
2545   }
2546   { author sort.format.names }

```

```

2547   if$
2548 }

```

`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.

```

2550 FUNCTION {presort}
2551 { calc.label
2552   label sortify
2553   " "
2554   *
2555   type$ "book" =
2556   type$ "inbook" =
2557   or
2558   'author.editor.sort
2559   { type$ "proceedings" =
2560     'editor.sort
2561     'author.sort
2562     if$
2563   }
2564   if$
2565   #1 entry.max$ substring$
2566   'sort.label :=
2567   sort.label
2568   *
2569   " "
2570   *
2571   title field.or.null
2572   sort.format.title
2573   *
2574   #1 entry.max$ substring$
2575   'sort.key$ :=
2576 }

```

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

```

2578 ITERATE {presort}
2579 SORT

```

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

```

2580 STRINGS { last.label next.extra }
2581 INTEGERS { last.extra.num last.extra.num.extended last.extra.num.blank number.label }
2582 FUNCTION {initialize.extra.label.stuff}
2583 { #0 int.to.chr$ 'last.label :=
2584   "" 'next.extra :=
2585   #0 'last.extra.num :=
2586   "a" chr.to.int$ #1 - 'last.extra.num.blank :=
2587   last.extra.num.blank 'last.extra.num.extended :=
2588   #0 'number.label :=
2589 }
2590 FUNCTION {forward.pass}
2591 { last.label label =
2592   { last.extra.num #1 + 'last.extra.num :=
2593     last.extra.num "z" chr.to.int$ >
2594     { "a" chr.to.int$ 'last.extra.num :=
2595       last.extra.num.extended #1 + 'last.extra.num.extended :=
2596     }
2597     'skip$
2598   if$
2599   last.extra.num.extended last.extra.num.blank >
2600   { last.extra.num.extended int.to.chr$
2601     last.extra.num int.to.chr$
2602     * 'extra.label := }
2603   { last.extra.num int.to.chr$ 'extra.label := }
2604   if$
2605 }
2606 { "a" chr.to.int$ 'last.extra.num :=
2607   "" 'extra.label :=
2608   label 'last.label :=
2609 }

```

```

2610     if$
2611     number.label #1 + 'number.label :=
2612 }
2613 FUNCTION {reverse.pass}
2614 { next.extra "b" =
2615   { "a" 'extra.label := }
2616   'skip$
2617   if$
2618   extra.label 'next.extra :=
2619   extra.label
2620   duplicate$ empty$
2621   'skip$
2622   { "{\natexlab{" swap$ * "}" * }
2623   if$
2624   'extra.label :=
2625   label extra.label * 'label :=
2626 }

```

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.

```

2627 EXECUTE {initialize.extra.label.stuff}
2628 ITERATE {forward.pass}
2629 REVERSE {reverse.pass}
2630 FUNCTION {bib.sort.order}
2631 { sort.label
2632   " "
2633   *
2634   year field.or.null sortify
2635   *
2636   " "
2637   *
2638   title field.or.null
2639   sort.format.title
2640   *
2641   #1 entry.max$ substring$
2642   'sort.key$ :=
2643 }
2644 ITERATE {bib.sort.order}
2645 SORT

```

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

```

2646 FUNCTION {begin.bib}
2647 { preamble$ empty$
2648   'skip$
2649   { preamble$ write$ newline$ }
2650   if$
2651   "\begin{thebibliography}{ number.label int.to.str$ * "}" *
2652   write$ newline$
2653   "\providecommand{\natexlab}[1]{#1}"
2654   write$ newline$
2655   "\providecommand{\url}[1]{\texttt{#1}}"
2656   write$ newline$
2657   "\providecommand{\urlprefix}{Available from: }"
2658   write$ newline$
2659   "\providecommand{\urldateprefix}{Accessed }"
2660   write$ newline$
2661   "\providecommand{\selectlanguage}[1]{\relax}"
2662   write$ newline$
2663   "\providecommand{\bibinfo}[2]{#2}"
2664   write$ newline$
2665   "\providecommand{\eprint}[2][\url{#2}]"
2666   write$ newline$
2667 }
2668 EXECUTE {begin.bib}
2669 EXECUTE {init.state.consts}

```

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

```
2670 ITERATE {call.type$}
```

After the last entry, we close the L^AT_EX environment and stop writing to the .bbl file.

```
2671 FUNCTION {end.bib}  
2672 { newline$  
2673   "\end{thebibliography}" write$ newline$  
2674 }  
2675 EXECUTE {end.bib}
```