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

Maintainer: Alex Ball[*]

Package v6.0 – 25 May 2022

1 Introduction

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

1.1 Installation

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

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

1.1.1 Managed way

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

1.1.2 Automated way

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

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

---

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

1.1.3 Manual way

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

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

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

   If you are using MikTeX, consult the MikTeX manual entry for integrating local additions. You can use one of the roots (TeX trees) already defined – preferably one of the User roots – or set up a new one.
4. Move the files to your TeX tree as follows:
   - `source/bibtex/bath-bst`: `bath-bst.dtx`, (`bath-bst.ins`)
   - `bibtex/bst/bath-bst`: `bath.bst`, `bathx.bst`
   - `doc/bibtex/bath-bst`: `bath-bst.pdf`, `bath-bst-v1.pdf`, `README.md`
5. You may then have to update your installation's file name database before TeX and friends can see the files.

2 Choosing which variant to use

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

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

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

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

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

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

3 Using the style

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

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

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

```
\bibliography{file}
```

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

```
While collections can be supplemented by other means \citep{devlin.etal2021ipp},
the absence of an invisible collection amongst historians is noted by
\citet[p.556]{stieg1981inh}. It may be, as \citet{burchard1965hhl} points out,
that they have no assistants or are reluctant to delegate, or it may be
down to economic factors \citep{adams2009tc1, adams2014tc2, gb.pa2014,
adams2017tc3}\dots
```

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

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

4 Breaking changes between versions

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

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

4.1 Version 3

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

4.2 Version 5

- In the 2021 revision of the style, the titles of broadcast media (TV, radio) became upright, with series information for titled episodes being italic. To get this new formatting, you must use the `type` field as shown.

- Series names (except those in `legislation` and `jurisdiction` entries) are now coerced to sentence case, so any capital letters you want to keep need to be protected with braces.
- The '[Online]' tag is now also triggered by `urldate`/`urlyear`. In cases where it occurs directly after the title, it is no longer left behind (after the year) if the title is moved to the head of the reference.
- The `nameaddon` field is now supported for annotating author lists with, say, a social media handle.
- The `article` and `manual` entry types now support the `month` field.
- Maps need to be given the `book` entry type; previously `manual` was recommended.

4.3 Version 6

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

5 Examples

The examples below are shown in three parts. The first, marked with 📖, shows an extract from the *Referencing guide: Harvard Bath* or *Referencing images*. The second, marked with ⚙, shows the reference as formatted by BibTeX. 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

ℹ 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.hbcpnetbase.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.hbcpnetbase.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.hbcpnetbase.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

ℹ 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

ℹ 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

ℹ 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.

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

ℹ 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-coron
  urlyear = {30 April 2020}}
```

RX

- 💡 If citing an article yet to be officially published, use the `pubstate` field (from biblatex) with the keyword `inpress` for ‘in press’ and `inpreparation` or `submitted` (whichever is more accurate) for ‘preprint’.

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

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

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

RX

Preprint or trial study in a digital repository

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

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

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

RX

- 💡 The Harvard (Bath) style treats preprints as grey literature, and so the `techreport` entry type (or its alias `report`) is the best match semantically.

- 💡 Use `library` to specify the name of the preprint repository. Note that the workaround in `bath.bst` of using `note` instead does not work in `bathx.bst`.

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

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

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

RX

Newspaper article

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

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

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

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

RX

ℹ The extra braces around the `month` field aren't necessary with BibTeX, but are needed if you want compatibility with biblatex.

📕 The Independent, 1992. Picking up the bills. *The Independent*, 4 June, p.28a.

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

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

RX

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

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

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

RX

5.3 Conference papers

ℹ 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

ℹ 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

ℹ 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`.

ℹ 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

ℹ 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

ℹ Internally, `patent` is an alias for `manual`.

Unpublished written material and personal communications

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

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

RX

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

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

RX

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

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

RX

5.5 Audiovisual materials

Image

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

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

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

RI

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

📕 Iliff, D., 2006. *Royal Crescent in Bath, England - July 2006* [Online]. San Francisco: Wikimedia Foundation. Available from: `https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg` [Accessed 7 January 2016].

⚙ Iliff, D., 2006. *Royal Crescent in Bath, England - July 2006* [Online]. San Francisco: Wikimedia Foundation. Available from: `https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg` [Accessed 7 January 2016].

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

RI

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

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

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

RI

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

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

💡 Put the clarification of the resource type in `entrysubtype`. In most cases, you can use `titleaddon` instead, but `entrysubtype` will correctly suppress ‘Online’ being shown while `titleaddon` will not.

ℹ The `@image` entry type can be used for any visual artistic work, including sculptures and interactive exhibits.

Map

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

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

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

RX

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

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

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

RX

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

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

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

RX

Film, video or DVD

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

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

RX

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

📘 *The elephant man* [Online], 1980. Film. Directed by David Lynch. USA: Brooksfilms. Available from: BBC iPlayer [Accessed 4 May 2021].
⚙ *The elephant man* [Online], 1980. Film. Directed by David Lynch. USA: Brooksfilms. 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 = {Brooksfilms},
  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

ℹ 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{beethoven1950symph1,
  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

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

Computer program

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

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

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

RX

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

💡 Put the clarification of the resource type in `entrysubtype`. In most cases, you can use `titleaddon` instead, but `titleaddon` is shown immediately after the title and may be shown alongside the ‘Online’ label, whereas `entrysubtype` is printed after the `version` and supersedes the ‘Online’ label.

5.7 Works in languages other than English

Work in translation

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

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

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

RX

Work in the Roman alphabet

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

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

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

RX

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

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

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

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

RX

Work in a non-Roman alphabet

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

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

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

RX

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

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

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

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

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

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

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

RX

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

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

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

RX

5.8 Legal or government document: UK parliamentary reports/ legislation

House of Commons paper

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

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

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

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

RX

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

House of Lords paper

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

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

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

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

RX

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

Command paper

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

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

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

RX

House of Commons/House of Lords bill

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

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

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

RX

Act of Parliament (UK Statutes) before 1963

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

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

RX

Act of Parliament (UK Statutes) 1963 onwards

📕 *Pensions Act 2014*, c.19. London: TSO.
⚙ *Pensions Act 2014*, c.19. London: TSO.

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

RX

Statutory instrument

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

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

RX

💡 Use the `entrysubtype` ‘secondary’ to put the number in the right place.

5.9 Legal or government document: EU reports/legislation

EU publication

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

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

RX

EU regulation or directive, decision, recommendation or opinion

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

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

RX

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

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

(Council Regulation [EC] 1984/2003)

5.10 Legal or government document: case reports

Legal case study

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

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

RX

ℹ 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 biblatex have been emulated rather crudely. You can only supply one of each and the text must match exactly (so don't add spaces or anything like that).

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

```
title = {Seldon v.~{Clarkson} {Wright} \& {Jakes}},
sortyear = {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 `casenmuber` field) for the case number. For Commission Decisions, use the (non-standard) `casenumber` or (biblatex-oscola) `userb` field for the Commission case number, `number` for the formal decision number, and give 'Commission' as the `institution`.

References

Adams, G., 2009. *Test citation one.* London: Imperial College Bookstall.

Adams, G., 2014. *Test citation two.* Oxford: Oxford University Press.

Adams, G., 2017. *Test citation three.* Cambridge: Cambridge University Press.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Council Regulation (EC) 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 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].

The elephant man [Online], 1980. Film. Directed by David Lynch. USA: Brooksfilms. 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/uksi/2012/1916/pdfs/uksi_20121916_en.pdf` [Accessed 17 April 2016].

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

Iliff, D., 2006. *Royal Crescent in Bath, England - July 2006* [Online]. San Francisco: Wikimedia Foundation. Available from: `https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg` [Accessed 7 January 2016].

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

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

Macbeth, 1948. Film. Directed by Orson Welles. USA: Republic Pictures.

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

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

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

Newman, R., 2010. Malaria control beyond 2010. *Brit. Med. J.*, 341(7765), pp.157–208.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

University of Bath Library [@unibathlib], 2021. *Instagram post* [Online], 15 March. Available from: `https://www.instagram.com/p/CMb5GHjFv9A/?utm_source=ig_web_copy_link` [Accessed 7 May 2021].

Wetzstein, G., Ozcan, A., Gigan, S., Fan, S., Englund, D., 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

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 BibTEX 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.

```
ENTRY
  { address
    archive
    author
    booktitle
    casenumber
    chapter
    doi
    edition
    editor
    eid
    entrysubtype
    eprint
    eventyear
    howpublished
    institution
    journal
    key
    keywords
    language
    library
    month
    nameaddon
    note
    number
    options
    organization
    pages
    publisher
    pubstate
    school
    series
    sortyear
    title
    titleaddon
    translator
    type
    url
    urldate
    urlyear
    userb
    venue
    version
    volume
    year
  }
  {}
  { label extra.label sort.label short.list }
```

7.2 Writing out references to a file

This is the first of three sections that deal with writing out to the .bbl bibliography file. That file contains the `thebibliography` environment, which in turn contains `\bibitem` commands each providing citation data and followed by a formatted reference. The business of writing the environment, sorting the items and calculating the first half of the citation data for each item is covered in section 7.8. Calculating the second half of the citation data and writing the `\bibitem` command is covered in section 7.5; some of the functions defined there are used in section 7.8, hence the backward structure. This section contains generic functions for writing out the formatted reference.

We define a set of integers for saving an output state. We will use them to 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.

```
INTEGERS {
  output.state before.all mid.clause mid.sentence after.sentence after.block
  online.shown
}
FUNCTION {init.state.consts}
{ #0 'before.all :=
  #1 'mid.clause :=
  #2 'mid.sentence :=
  #3 'after.sentence :=
  #4 'after.block :=
}
```

We also define two variables to store strings in.

```
STRINGS { s t }
```

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

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

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

```
FUNCTION {output.nonnull}
{ 's :=
  output.state mid.sentence =
    { ", " * write$ }
    { output.state after.sentence =
        { add.period$ " " * write$ }
        { output.state before.all =
            'write$
            { output.state mid.clause =
                { " " * write$ }
                { add.period$ write$
                  newline$
                  "\newblock " write$
                }
              if$
            }
          if$
        }
      if$
      mid.sentence 'output.state :=
    }
  if$
  s
}
```

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’.

```
FUNCTION {end.clause}
{ output.state mid.clause =
    { mid.sentence 'output.state := }
    'skip$
  if$
}
```

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

```
FUNCTION {output}
{ duplicate$ empty$
    { pop$ end.clause }
    'output.nonnull
  if$
}
FUNCTION {output.check}
{ 't :=
  duplicate$ empty$
    { pop$ "empty " t * " in " * cite$ * warning$
      end.clause
    }
    'output.nonnull
  if$
}
```

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.

```
FUNCTION {new.block}
{ output.state before.all =
    'skip$
    { after.block 'output.state := }
  if$
}
FUNCTION {new.sentence}
{ output.state after.block =
    'skip$
    { output.state before.all =
        'skip$
        { after.sentence 'output.state := }
      if$
    }
  if$
}
FUNCTION {continue.clause}
{ output.state after.sentence =
    'skip$
    { output.state after.block =
        'skip$
        { output.state before.all =
            'skip$
            { mid.clause 'output.state := }
          if$
        }
      if$
    }
  if$
}
```

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

```
FUNCTION {date.block}
{ new.block
}
```

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

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

```
FUNCTION {bibinfo.check}
{ swap$
  duplicate$ missing$
    { pop$ pop$
      ""
    }
    { duplicate$ empty$
        { swap$ pop$
        }
        { swap$
          "\bibinfo{" swap$ * "}{" * swap$ * "}" *
        }
      if$
    }
  if$
}
FUNCTION {bibinfo.warn}
{ swap$
  duplicate$ missing$
    { swap$ "missing " swap$ * " in " * cite$ * warning$ pop$
      ""
    }
    { duplicate$ empty$
        { swap$ "empty " swap$ * " in " * cite$ * warning$
        }
        { swap$
          "\bibinfo{" swap$ * "}{" * swap$ * "}" *
        }
      if$
    }
  if$
}
```

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.

```
FUNCTION {fin.entry}
{ add.period$
  write$
  newline$
}
```

7.3 Convenience functions

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

```
FUNCTION {not}
{   { #0 }
    { #1 }
  if$
}
FUNCTION {and}
{   'skip$
    { pop$ #0 }
  if$
}
FUNCTION {or}
{   { pop$ #1 }
    'skip$
  if$
}
```

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

```
FUNCTION {new.block.checkb}
{ empty$
  swap$ empty$
  and
    'skip$
    'new.block
  if$
}
```

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

```
FUNCTION {field.or.null}
{ duplicate$ empty$
    { pop$ "" }
    'skip$
  if$
}
```

The `emphasize` function applies emphasis to the topmest token.

```
FUNCTION {emphasize}
{ duplicate$ empty$
    { pop$ "" }
    { "\emph{" swap$ * "}" * }
  if$
}
```

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.

```
FUNCTION {tie.or.space.prefix}
{ duplicate$ text.length$ #3 <
    { "~" }
    { " " }
  if$
  swap$
}
```

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

```
FUNCTION {capitalize}
{ "u" change.case$ "t" change.case$ }
```

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

```
FUNCTION {space.word}
{ " " swap$ * " " * }
```

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

```
FUNCTION {either.or.check}
{ empty$
    'pop$
    { "can't use both " swap$ * " fields in " * cite$ * warning$ }
  if$
}
```

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

```
FUNCTION {is.online}
{ online.shown
    { #0 }
    { url empty$
      doi empty$ and
      urldate empty$ and
      urlyear empty$ and
        { type$ "online" =
            { #1 }
            { #0 }
          if$
        }
        { #1 }
      if$
    }
  if$
}
```

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.

```
FUNCTION {bbl.and}
{ "and"}

FUNCTION {bbl.byeditor}
{ "Ed. by" }

FUNCTION {bbl.etal}
{ "et~al." }

FUNCTION {bbl.editors}
{ "eds" }

FUNCTION {bbl.editor}
{ "ed." }

FUNCTION {bbl.edby}
{ "edited by" }

FUNCTION {bbl.translator}
{ "Trans." }

FUNCTION {bbl.edition}
{ "ed." }

FUNCTION {bbl.volume}
{ "vol." }

FUNCTION {bbl.of}
{ "of" }

FUNCTION {bbl.number}
{ "no." }

FUNCTION {bbl.nr}
{ "no." }

FUNCTION {bbl.in}
{ "in" }

FUNCTION {bbl.pages}
{ "pp." }

FUNCTION {bbl.page}
{ "p." }

FUNCTION {bbl.chapter}
{ "chap." }

FUNCTION {bbl.techrep}
{ "Tech. Rep." }

FUNCTION {bbl.mthesis}
{ "Master's thesis" }

FUNCTION {bbl.phdthesis}
{ "Ph.D. thesis" }

FUNCTION {bbl.first}
{ "1st" }

FUNCTION {bbl.second}
{ "2nd" }

FUNCTION {bbl.third}
{ "3rd" }

FUNCTION {bbl.fourth}
{ "4th" }
```

```

FUNCTION {bbl.fifth}
{ "5th" }

FUNCTION {bbl.st}
{ "st" }

FUNCTION {bbl.nd}
{ "nd" }

FUNCTION {bbl.rd}
{ "rd" }

FUNCTION {bbl.th}
{ "th" }

FUNCTION {bbl.commission}
{ "Commission Decision" }

FUNCTION {bbl.online}
{ "[Online]" }

FUNCTION {bbl.nodate}
{ "n.d." }

FUNCTION {bbl.inpreparation}
{ "preprint" }

FUNCTION {bbl.submitted}
{ "preprint" }

FUNCTION {bbl.inpress}
{ "in press" }

FUNCTION {bbl.unpublished}
{ "Unpublished" }
```

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

```
MACRO {jan} {"January"}

MACRO {feb} {"February"}

MACRO {mar} {"March"}

MACRO {apr} {"April"}

MACRO {may} {"May"}

MACRO {jun} {"June"}

MACRO {jul} {"July"}

MACRO {aug} {"August"}

MACRO {sep} {"September"}

MACRO {oct} {"October"}

MACRO {nov} {"November"}

MACRO {dec} {"December"}
```

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

```
FUNCTION {eng.ord}
{ duplicate$ "1" swap$ *
  #-2 #1 substring$ "1" =
     { bbl.th * }
     { duplicate$ #-1 #1 substring$
       duplicate$ "1" =
         { pop$ bbl.st * }
         { duplicate$ "2" =
             { pop$ bbl.nd * }
             { "3" =
                 { bbl.rd * }
                 { bbl.th * }
               if$
             }
           if$
          }
       if$
     }
   if$
}
```

`word.in` provides the ‘In:’ before book titles.

```
  FUNCTION {word.in}
{ bbl.in capitalize
  ":" *
  " " * }
```

7.5 Writing out citation data to a file

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

```
INTEGERS { nameptr namesleft numnames }
STRINGS  { bibinfo }
FUNCTION {format.names}
{ 'bibinfo :=
  duplicate$ empty$
    'skip$
    { 's :=
      "" 't :=
      #1 'nameptr :=
      s num.names$ 'numnames :=
      numnames 'namesleft :=
        { namesleft #0 > }
        { s nameptr
          "{ll}{, jj}{, f{.}.}{~vv}"
          format.name$
          bibinfo bibinfo.check
          't :=
          nameptr #1 >
            { namesleft #1 >
                { ", " * t * }
                { s nameptr "{ll}" format.name$ duplicate$ "others" =
                    { 't := }
                    { pop$ }
                  if$
                  t "others" =
                    { " " * bbl.etal *
                    }
                    { bbl.and
                      space.word * t *
                    }
                  if$
                }
              if$
            }
            't
          if$
          nameptr #1 + 'nameptr :=
          namesleft #1 - 'namesleft :=
        }
      while$
    }
  if$
}
```

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

```
FUNCTION {format.names.ed}
{ 'bibinfo :=
  duplicate$ empty$
    'skip$
    { 's :=
      "" 't :=
      #1 'nameptr :=
      s num.names$ 'numnames :=
      numnames 'namesleft :=
        { namesleft #0 > }
        { s nameptr
          "{f{.}.~}{vv~}{ll}{ jj}"
          format.name$
          bibinfo bibinfo.check
          't :=
          nameptr #1 >
            { namesleft #1 >
                { ", " * t * }
                { s nameptr "{ll}" format.name$ duplicate$ "others" =
                    { 't := }
                    { pop$ }
                  if$
                  t "others" =
                    { " " * bbl.etal *
                    }
                    { bbl.and
                      space.word * t *
                    }
                  if$
                }
              if$
            }
            't
          if$
          nameptr #1 + 'nameptr :=
          namesleft #1 - 'namesleft :=
        }
      while$
    }
  if$
}
```

`format.authors` returns author names.

```
FUNCTION {format.authors}
{ author "author" format.names
}
```

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

```
FUNCTION {format.byeditors}
{ editor "editor" format.names.ed duplicate$ empty$ 'skip$
    { bbl.byeditor
      " " *
      swap$ *
    }
  if$
}
```

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

```
FUNCTION {get.bbl.editor}
{ editor num.names$ #1 > 'bbl.editors 'bbl.editor if$ }
FUNCTION {format.editors}
{ editor "editor" format.names duplicate$ empty$ 'skip$
    { "," *
      " " *
      get.bbl.editor
      *
    }
  if$
}
```

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

```
FUNCTION {format.translators}
{ translator "translator" format.names.ed duplicate$ empty$ 'skip$
    { "," *
      " " *
      bbl.translator *
      "(" swap$ *
      ")" *
    }
  if$
}
```

`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.

```
FUNCTION {format.full.names}
{'s :=
 "" 't :=
  #1 'nameptr :=
  s num.names$ 'numnames :=
  numnames 'namesleft :=
    { namesleft #0 > }
    { s nameptr
      "{vv~}{ll}" format.name$
      't :=
      nameptr #1 >
        { namesleft #1 >
            { ", " * t * }
            { s nameptr "{ll}" format.name$ duplicate$ "others" =
                { 't := }
                { pop$ }
              if$
              t "others" =
                { " " * bbl.etal *
                }
                { bbl.and
                  space.word * t *
                }
              if$
            }
          if$
        }
        't
      if$
      nameptr #1 + 'nameptr :=
      namesleft #1 - 'namesleft :=
    }
  while$
}
```

`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.

```
FUNCTION {author.editor.key.full}
{ author empty$
    { editor empty$
        { key empty$
            { cite$ #1 #3 substring$ }
            'key
          if$
        }
        { editor format.full.names }
      if$
    }
    { author format.full.names }
  if$
}
```

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

```
FUNCTION {editor.key.full}
{ editor empty$
    { key empty$
        { cite$ #1 #3 substring$ }
         'key
      if$
    }
    { editor format.full.names }
  if$
}
```

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

```
FUNCTION {author.key.full}
{ author empty$
    { key empty$
        { cite$ #1 #3 substring$ }
         'key
      if$
    }
    { author format.full.names }
  if$
}
```

`make.full.names` chooses between the above.

```
FUNCTION {make.full.names}
{ type$ "book" =
  type$ "inbook" =
  or
    'author.editor.key.full
    { type$ "proceedings" =
        'editor.key.full
        'author.key.full
      if$
    }
  if$
}
```

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

```
FUNCTION {output.bibitem}
{ newline$
  "\bibitem[{" write$
  label write$
  ")" make.full.names duplicate$ short.list =
     { pop$ }
     { * }
   if$
  "}]{" * write$
  cite$ write$
  "}" write$
  newline$
  ""
  #0 'online.shown :=
  before.all 'output.state :=
}
```

`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.

```
FUNCTION {format.key}
{ empty$
```

```
    { key field.or.null }
    { "" }
  if$
  nameaddon duplicate$ empty$
    'pop$
    { swap$ duplicate$ empty$
        'skip$
        { " " * }
      if$
      swap$ "[" swap$ * * "]" *
      continue.clause
    }
  if$
}
```

7.6 Field formats

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

7.6.1 Title block

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

```
FUNCTION {select.language}
{ duplicate$ empty$
    'pop$
    { language empty$
        'skip$
        { "{\selectlanguage{" language * "}" * swap$ * "}" * }
      if$
    }
  if$
}
```

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

```
FUNCTION {format.title}
{ title
  duplicate$ empty$ 'skip$
    { "t" change.case$ }
  if$
  "title" bibinfo.check
  duplicate$ empty$
    'skip$
    { select.language }
  if$
}
```

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

```
FUNCTION {format.btitle}
{ title
  duplicate$ empty$ 'skip$
    { "t" change.case$ }
  if$
  "title" bibinfo.check
  duplicate$ empty$
    'skip$
    { emphasize select.language }
  if$
}
```

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

```
FUNCTION {format.avtitle}
{ title
  duplicate$ empty$ 'skip$
    { "t" change.case$ }
  if$
  "title" bibinfo.check
  duplicate$ empty$
    'skip$
    { type duplicate$ empty$
        { pop$ emphasize }
        { duplicate$ "TV" = swap$ "Radio" = or
            'skip$
            'emphasize
          if$
        }
      if$
      select.language
    }
  if$
}
```

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

```
FUNCTION {format.titleaddon}
{ titleaddon
  duplicate$ empty$ 'skip$
    { "titleaddon" bibinfo.check }
  if$
}
```

This is the `bathx.bst` version:

```
FUNCTION {format.titleaddon}
{ titleaddon
  duplicate$ empty$ 'skip$
    { "[" swap$ "titleaddon" bibinfo.check * "]" * }
  if$
}
```

`format.version` returns the version field.

```
FUNCTION {format.version}
{ version
  duplicate$ empty$ 'skip$
    { "(v." swap$ *
      "version" bibinfo.check
      ")" *
    }
  if$
}
```

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

```
FUNCTION {print.labeltitle.addenda}
{ continue.clause
```

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

```
  online.shown
    'skip$
    { format.version output
      continue.clause
      format.titleaddon output
      #1 'online.shown :=
    }
  if$
```

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

```
  format.titleaddon output
  continue.clause
  type$ duplicate$ "legislation" = not
  swap$ "jurisdiction" = not
  and
    { format.version output
      is.online
        { entrysubtype duplicate$ empty$
            { pop$
              journal empty$
                booktitle empty$
                  type$ "inreference" =
                  author empty$
                  and
                or
              and
                library empty$
                type$ duplicate$ "image" =
                swap$ duplicate$ "audio" =
                swap$ duplicate$ "video" =
                swap$ duplicate$ "music" =
                swap$ "movie" =
                or
                or
                or
                or
                or
              and
                series empty$
                  type field.or.null
                  duplicate$ "TV" =
                  swap$ "Radio" =
                  or not
                or
              and
                { continue.clause
                  bbl.online output
                  #1 'online.shown :=
                }
                'skip$
              if$
            }
            { "entrysubtype" bibinfo.check
              "[" swap$ * "]" *
              continue.clause output
            }
          if$
        }
        'skip$
      if$
    }
    'skip$
  if$
```

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

```
}
FUNCTION {print.title.addenda}
{ continue.clause
```

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

```
  online.shown
    'skip$
    { format.version output
      continue.clause
      format.titleaddon output
      #1 'online.shown :=
    }
  if$
```

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

```
  type$ duplicate$ "legislation" = not
  swap$ "jurisdiction" = not
  and
    { format.titleaddon output
      continue.clause
    }
    'skip$
  if$
  format.version output
  is.online
    { entrysubtype duplicate$ empty$
        type$ duplicate$ "legislation" =
        swap$ "jurisdiction" =
        or
      or
        { pop$
          journal empty$
            booktitle empty$
              type$ "inreference" =
              author empty$
              and
            or
          and
            library empty$
            type$ duplicate$ "image" =
            swap$ duplicate$ "audio" =
            swap$ duplicate$ "video" =
            swap$ duplicate$ "music" =
            swap$ "movie" =
            or
            or
            or
            or
            or
          and
            series empty$
              type field.or.null
              duplicate$ "TV" =
              swap$ "Radio" =
              or not
            or
          and
            { continue.clause
              bbl.online output
              #1 'online.shown :=
            }
            'skip$
          if$
        }
        { "entrysubtype" bibinfo.check
          "[" swap$ * "]" *
          continue.clause output
        }
      if$
    }
    'skip$
  if$
```

We finish off with editor/translator information.

```
  continue.clause
  type$ "reference" =
    { format.byeditors output }
    'skip$
  if$
  format.translators output
}
```

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

```
FUNCTION {format.label.booktitle}
{ booktitle
  duplicate$ empty$ 'skip$
```

```
    { "t" change.case$ }
  if$
  "booktitle" bibinfo.check
}
```

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

```
FUNCTION {format.booktitle}
{ format.label.booktitle
  emphasize
}
```

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

```
FUNCTION {format.in.ed.booktitle}
{ format.booktitle
  duplicate$ empty$
    'skip$
    {
```

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

```
      is.online
        { " " *
          bbl.online *
          #1 'online.shown :=
        }
        'skip$
      if$
```

```
      editor "editor" format.names.ed duplicate$ empty$ 'pop$
        { "," *
          " " *
          get.bbl.editor add.period$
          " " *
          * swap$
          *
          word.in swap$ *
        }
      if$
    }
  if$
}
```

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

```
FUNCTION {legal.journal.shape}
{ duplicate$ "OJ" =
    'emphasize
    'skip$
  if$
}
```

7.6.2 Dates

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

```
FUNCTION {nodate.check}
{
  skip$
}
```

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.

```
FUNCTION {nodate.check}
{
duplicate$ empty$
  sortyear empty$
  and
    { options field.or.null "nonodate" =
        'skip$
        { pop$ bbl.nodate }
      if$
    }
    'skip$
  if$
}
```

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

```
FUNCTION {format.date}
{ year "year" bibinfo.check nodate.check
  extra.label *
}
```

`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.

```
FUNCTION {format.date.brackets}
{ format.date
  "[" swap$ * "]" *
}
FUNCTION {format.date.parens}
{ format.date
  "(" swap$ * ")" *
}
FUNCTION {format.jur.date.essential}
{ options field.or.null "scottish-style" =
  keywords field.or.null "sc" =
  or
    'format.date
    'format.date.brackets
  if$
}
FUNCTION {format.jur.date}
{ journal field.or.null 's :=
  s "OJ" =
  s "ECR" =
  or
    'format.date.brackets
    { options field.or.null "year-essential=false" =
        'format.date.parens
        { volume empty$
            'format.jur.date.essential
            { options field.or.null "year-essential=true" =
                'format.jur.date.essential
                'format.date.parens
              if$
            }
          if$
        }
      if$
    }
  if$
}
```

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

```
FUNCTION {format.eu.date}
{ journal field.or.null 's :=
  s "OJ" =
  s "ECR" =
  or
    'format.date.brackets
    'format.date
  if$
}
```

7.6.3 Series, types and numbering

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

```
FUNCTION {format.series}
{ series
  duplicate$ empty$ 'skip$
    { "t" change.case$ }
  if$
  "series" bibinfo.check
}
```

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

```
FUNCTION {format.bvolume}
{ volume empty$
    { "" }
    { bbl.volume volume tie.or.space.prefix
      "volume" bibinfo.check * *
      format.series
      duplicate$ empty$ 'pop$
        { emphasize ", " * swap$ * }
      if$
      "volume and number" number either.or.check
    }
  if$
}
```

`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.

```
FUNCTION {format.number.series}
{ volume empty$
    { number empty$
        { series field.or.null }
        { series empty$
            { number "number" bibinfo.check }
            { format.series
              number tie.or.space.prefix "number" bibinfo.check * *
            }
          if$
        }
      if$
    }
    { "" }
  if$
}
```

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

```
FUNCTION {format.series.episode}
{ series
  duplicate$ empty$
    'skip$
    { "t" change.case$ }
  if$
  "series" bibinfo.check emphasize
  number "number" bibinfo.check emphasize
  duplicate$ empty$
    { * }
    { swap$
      duplicate$ empty$
        { * }
        { ", " * swap$ * }
      if$
    }
  if$
}
```

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

```
FUNCTION {is.num}
{ chr.to.int$
  duplicate$ "0" chr.to.int$ < not
  swap$ "9" chr.to.int$ > not and
}
```

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

```
FUNCTION {extract.num}
{ duplicate$ 't :=
  "" 's :=
  { t empty$ not }
  { t #1 #1 substring$
    t #2 global.max$ substring$ 't :=
    duplicate$ is.num
      { s swap$ * 's := }
      { pop$ "" 't := }
    if$
  }
  while$
  s empty$
    'skip$
    { pop$ s }
  if$
}
```

`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.

```
FUNCTION {convert.edition}
{ extract.num "l" change.case$ 's :=
  s "first" = s "1" = or
    { bbl.first 't := }
    { s "second" = s "2" = or
        { bbl.second 't := }
        { s "third" = s "3" = or
            { bbl.third 't := }
            { s "fourth" = s "4" = or
                { bbl.fourth 't := }
                { s "fifth" = s "5" = or
                    { bbl.fifth 't := }
                    { s #1 #1 substring$ is.num
                        { s eng.ord 't := }
                        { edition 't := }
                      if$
                    }
                  if$
                }
              if$
            }
          if$
        }
      if$
    }
  if$
  t
}
```

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

```
FUNCTION {format.edition}
{ edition duplicate$ empty$ 'skip$
    { convert.edition
      output.state mid.sentence =
        { "l" }
        { "t" }
      if$ change.case$
      "edition" bibinfo.check
      " " * bbl.edition *
    }
  if$
}
```

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.

```
FUNCTION {format.thesis.type}
{ type duplicate$ empty$
    'pop$
    { swap$ pop$
      "t" change.case$ "type" bibinfo.check
    }
  if$
}
```

`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', 'Cmnd' or 'Cm', it is followed by a dot rather than a comma (implemented as `add.cmd.number`). To help with this, `format.legr.number` consumes one token (the number) and returns the series and number as one token.

```
FUNCTION {add.cmd.number}
{ ". " swap$ "number" bibinfo.check *
}
FUNCTION {format.legr.number}
{ series field.or.null 's :=
  s "C" =
    'add.cmd.number
    { s "Cd" =
        'add.cmd.number
        { s "Cmd" =
            'add.cmd.number
            { s "Cmnd" =
                'add.cmd.number
                { s "Cm" =
                    'add.cmd.number
                    { s purify$ #1 #3 substring$ 't :=
                      t "HL " =
                        { ", (" swap$
                          "number" bibinfo.check *
                          ")" *
                        }
                        { ", " swap$
                          "number" bibinfo.check *
                        }
                      if$
                    }
                  if$
                }
              if$
            }
          if$
        }
      if$
    }
  if$
}
FUNCTION {format.tr.number}
{ series duplicate$ empty$
    { pop$
      type duplicate$ empty$
        { pop$
          number duplicate$ empty$
            { pop$ ""
            }
            { "(" swap$
              "number" bibinfo.check
              * ")" *
            }
```

```
     if$
    }
    { "(" swap$
     "t" change.case$ "type" bibinfo.check
     *
     number duplicate$ empty$
      'pop$
      { tie.or.space.prefix
       "number" bibinfo.check
       * *
      }
     if$
     ")" *
    }
   if$
  }
  { "t" change.case$ "(" swap$ *
   type duplicate$ empty$
    { pop$
     number duplicate$ empty$
      'pop$
      { format.legr.number * }
     if$
    }
    { "t" change.case$ "type" bibinfo.check
     *
     number duplicate$ empty$
      'pop$
      { tie.or.space.prefix
       "number" bibinfo.check
       * *
      }
     if$
    }
   if$
   ")" *
   continue.clause
  }
 if$
}
```

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

```
FUNCTION {format.manual.number}
{ series duplicate$ empty$
  { pop$
   type duplicate$ empty$
    { pop$
     number duplicate$ empty$
      { pop$ ""
      }
      { "number" bibinfo.check
      }
     if$
    }
    { "t" change.case$ "type" bibinfo.check
     number duplicate$ empty$
      'pop$
      { tie.or.space.prefix
       "number" bibinfo.check
       * *
      }
     if$
    }
   if$
  }
  { "t" change.case$
   number duplicate$ empty$
    { pop$
    }
    { tie.or.space.prefix
     "number" bibinfo.check
```

```
          * *
        }
      if$
      type duplicate$ empty$
        'pop$
        { ", " swap$ *
          "t" change.case$ "type" bibinfo.check
          *
        }
      if$
    }
  if$
}
```

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

```
FUNCTION {format.chapter}
{ chapter duplicate$ empty$
    { pop$ "" }
    { "chapter" bibinfo.check
      "c." swap$ *
    }
  if$
}
```

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

```
FUNCTION {format.series.number.chapter}
{ series duplicate$ empty$
    { pop$ "(" }
    { "series" bibinfo.check
      "(" swap$ *
    }
  if$
  type duplicate$ empty$
    { pop$
      number duplicate$ empty$
        'pop$
        { "number" bibinfo.check
          swap$ duplicate$ "(" =
            { swap$ * }
            { ", " * swap$ * }
          if$
        }
      if$
      chapter duplicate$ empty$
        'pop$
        { "chapter" bibinfo.check
          swap$ duplicate$ "(" =
            { swap$ * }
            { ", c." * swap$ * }
          if$
        }
      if$
    }
    { "type" bibinfo.check
      swap$ duplicate$ "(" =
        { swap$ * }
        { ", " * swap$ * }
      if$
      number duplicate$ empty$
        { pop$
          chapter duplicate$ empty$
            'pop$
            { "chapter" bibinfo.check
              " c." swap$ * *
            }
          if$
        }
        { "number" bibinfo.check
          " " swap$ * *
          chapter duplicate$ empty$
```

```
            'pop$
            { "chapter" bibinfo.check
              ", c." swap$ * *
            }
          if$
        }
      if$
    }
  if$
  ")" *
  duplicate$ "()" =
    { pop$ "" }
    'skip$
  if$
}
```

`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‘

```
FUNCTION {format.case.number}
{ casenumber duplicate$ empty$
    { pop$
      number duplicate$ empty$
        { pop$ "" }
        { institution field.or.null "Commission" =
          userb empty$ not
          and
            { userb "userb" bibinfo.check
              "(" swap$ * ") " * bbl.commission * " " *
              swap$ "number" bibinfo.check *
            }
            { "number" bibinfo.check
            "(" swap$ * ")" *
            }
          if$
        }
      if$
    }
    { "casenumber" bibinfo.check
      "(" swap$ * ")" *
      institution field.or.null "Commission" =
      number empty$ not
      and
        { " " * bbl.commission * " " *
          number "number" bibinfo.check *
        }
        'skip$
      if$
    }
  if$
}
```

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

```
FUNCTION {eu.case.check}
{ journal field.or.null "OJ" =
  number empty$ not
  or
  casenumber empty$ not
  or
    'continue.clause
    'new.block
  if$
}
```

7.6.4 Notes

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

```
FUNCTION {format.note}
{ note empty$
    { "" }
    { note #1 #1 substring$
      duplicate$ "{" =
        'skip$
        { output.state mid.sentence =
          output.state mid.clause =
          or
            { "l" }
            { "u" }
          if$
          change.case$
        }
      if$
      note #2 global.max$ substring$ * "note" bibinfo.check
    }
  if$
}
```

7.6.5 Publication block

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

```
FUNCTION {prepend.address}
{ 't :=
  ""
  address empty$ t empty$ and
    'skip$
    { address "address" bibinfo.check *
      t empty$
        'skip$
        { address empty$
            'skip$
            { ": " * }
          if$
          t *
        }
      if$
    }
  if$
}
```

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

```
FUNCTION {format.publisher.address}
{ publisher "publisher" bibinfo.warn prepend.address
}
FUNCTION {format.organization.address}
{ publisher empty$
    { organization "organization" bibinfo.check prepend.address }
    { publisher "publisher" bibinfo.warn prepend.address
      organization duplicate$ empty$
        'pop$
        { "organization" bibinfo.check ". " * swap$ * }
      if$
    }
  if$
}
FUNCTION {format.institution.address}
{ publisher empty$
    { institution "institution" bibinfo.check prepend.address }
    { publisher "publisher" bibinfo.warn prepend.address
      institution duplicate$ empty$
        'pop$
        { "institution" bibinfo.check ". " * swap$ * }
      if$
    }
  if$
}
```

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

```
FUNCTION {format.library}
{ library "library" bibinfo.check
  emphasize
```

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

```
  duplicate$ empty$ not
  is.online
  and
  journal empty$
  and
    { " " * bbl.online *
      #1 'online.shown :=
    }
    'skip$
  if$
```

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

```
}
FUNCTION {format.img.library}
{ library "library" bibinfo.check
}
FUNCTION {format.img.library.address}
{ library duplicate$ empty$
    { pop$
      institution duplicate$ empty$
        'skip$
        { "institution" bibinfo.check }
      if$
    }
    { "library" bibinfo.check }
  if$
  duplicate$ empty$
    { address "address" bibinfo.check * }
    { address duplicate$ empty$
        'pop$
        { "address" bibinfo.check
          swap$ ", " * swap$ *
        }
      if$
    }
  if$
}
FUNCTION {format.pub.org.lib.address}
{ publisher duplicate$ empty$
  { pop$
    organization duplicate$ empty$
      { pop$
        format.img.library.address }
      { "organization" bibinfo.check
        library empty$
          { prepend.address }
          { ". " *
            format.img.library.address *
          }
        if$
      }
    if$
  }
  { "publisher" bibinfo.warn prepend.address
    library empty$
      'skip$
      { ". " *
    format.img.library *
   }
  if$
 }
 if$
}
```

7.6.6 Pinpointing

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

```
INTEGERS { multiresult }
FUNCTION {multi.page.check}
{ 't :=
 #0 'multiresult :=
  { multiresult not
   t empty$ not
   and
  }
  { t #1 #1 substring$
   duplicate$ "-" =
   swap$ duplicate$ "," =
   swap$ "+" =
   or or
    { #1 'multiresult := }
    { t #2 global.max$ substring$ 't := }
   if$
  }
 while$
 multiresult
}
```

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

```
FUNCTION {n.dashify}
{ 't :=
 ""
  { t empty$ not }
  { t #1 #1 substring$ "-" =
    { t #1 #2 substring$ "--" = not
      { "--" *
       t #2 global.max$ substring$ 't :=
      }
      {   { t #1 #1 substring$ "-" = }
        { "-" *
         t #2 global.max$ substring$ 't :=
        }
       while$
      }
     if$
    }
    { t #1 #1 substring$ *
     t #2 global.max$ substring$ 't :=
    }
   if$
  }
 while$
}
```

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

```
FUNCTION {format.pages}
{ pages duplicate$ empty$ 'skip$
  { duplicate$ multi.page.check
    { bbl.pages swap$
     n.dashify
    }
    { bbl.page swap$
    }
   if$
   "pages" bibinfo.check
```

```
      *
    }
  if$
}
```

`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.

```
FUNCTION {format.journal.pages}
{ pages duplicate$ empty$ 'pop$
    { swap$ duplicate$ empty$
        { pop$ pop$ format.pages }
        { ", " *
          swap$
          n.dashify
          pages multi.page.check
            'bbl.pages
            'bbl.page
          if$
          swap$
          "pages" bibinfo.check
          * *
        }
      if$
    }
  if$
}
```

`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).

```
FUNCTION {format.journal.eid}
{ eid "eid" bibinfo.check
  duplicate$ empty$ 'pop$
    { swap$ duplicate$ empty$ 'skip$
      { ", " *
      }
      if$
      swap$ *
    }
  if$
}
```

`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.

```
FUNCTION {format.vol.num.pages}
{ pubstate field.or.null
  duplicate$ "inpress" =
    { pop$ bbl.inpress }
    { duplicate$ "submitted" =
        { pop$ bbl.submitted }
        { duplicate$ "inpreparation" =
            { pop$ bbl.inpreparation }
            { pop$ "" }
          if$
        }
      if$
    }
  if$
  duplicate$ empty$
    { pop$
      volume field.or.null
      duplicate$ empty$ 'skip$
        { "volume" bibinfo.check
        }
      if$
      number "number" bibinfo.check duplicate$ empty$ 'skip$
        { swap$ duplicate$ empty$
            { "there's a number but no volume in " cite$ * warning$ }
            'skip$
          if$
          swap$
          "(" swap$ * ")" *
        }
      if$ *
      month "month" bibinfo.check duplicate$ empty$
        'pop$
        { swap$ duplicate$ empty$
            'pop$
            { ", " * swap$ * }
          if$
        }
      if$
      eid empty$
        { format.journal.pages }
        { format.journal.eid }
      if$
    }
    'skip$
  if$
}
```

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

```
FUNCTION {format.chapter.pages}
{ chapter empty$
    'format.pages
    { type empty$
        { bbl.chapter }
        { type "l" change.case$
          "type" bibinfo.check
        }
      if$
      chapter tie.or.space.prefix
      "chapter" bibinfo.check
      * *
      pages empty$
        'skip$
        { ", " * format.pages * }
      if$
    }
  if$
}
```

`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.

```
FUNCTION {format.journal.series.vol.pages}
{ journal field.or.null
  duplicate$ "OJ" =
    { "journal" bibinfo.check
      emphasize
      series empty$
      volume empty$
      and
      pages empty$
      and
        'skip$
        { " " * }
      if$
      format.series
      volume "volume" bibinfo.check *
      duplicate$ empty$
        'skip$
        { pages field.or.null duplicate$ empty$
      'skip$
      { "/" swap$ * 
      }
     if$ *
    }
   if$ *
  }
  { duplicate$ "ECR" =
    { "journal" bibinfo.check
     volume "volume" bibinfo.check
     duplicate$ empty$
      'skip$
      { " " swap$ * "--" * * }
     if$
    }
    { "journal" bibinfo.check
```

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

```
     is.online
      { " " * bbl.online *
       #1 'online.shown :=
      }
      'skip$
     if$
```

```
     volume "volume" bibinfo.check
     duplicate$ empty$
      'pop$
      { " " * swap$ * }
     if$
     pages empty$
     eid empty$
     and
      'skip$
      { " " * }
     if$
    }
   if$
  }
 if$
 eid empty$
  { pages "pages" bibinfo.check * }
  { eid "eid" bibinfo.check * }
 if$
}
```

7.6.7 Access and URLs

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

```
FUNCTION {format.eprint}
{ eprint duplicate$ empty$
  'skip$
  { "\eprint"
   archive empty$
    'skip$
    { "[" * archive * "]" * }
   if$
   "{" * swap$ * "}" *
  }
 if$
}
```

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

```
FUNCTION {format.url}
{ doi empty$
  { url }
  { "https://doi.org/" doi * }
if$
duplicate$ empty$
  { pop$ "" }
  { "\urlprefix\url{" swap$ * "}" * }
if$
duplicate$ empty$
  urlyear empty$ not
  urldate empty$ not
  or
and
  { library duplicate$ empty$
      'pop$
      { "\urlprefix{}" swap$ * * }
    if$
  }
  'skip$
if$
urlyear empty$
  { urldate empty$
      'skip$
      { " [\urldateprefix{}" * urldate * "]" * }
    if$
  }
  { " [\urldateprefix{}" * urlyear * "]" * }
if$
}
```

7.6.8 Related items

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

```
FUNCTION {format.article.crossref}
{ word.in
  " \cite{" * crossref * "}" *
}
FUNCTION {format.incoll.inproc.crossref}
{ word.in
  " \cite{" * crossref * "}" *
}
```

`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.

```
FUNCTION {format.book.crossref}
{ volume duplicate$ empty$
    { "empty volume in " cite$ * "'s crossref of " * crossref * warning$
      pop$ word.in
    }
    { bbl.volume
      capitalize
      swap$ tie.or.space.prefix "volume" bibinfo.check * * bbl.of space.word *
    }
  if$
  " \cite{" * crossref * "}" *
}
```

7.7 Drivers for formatting specific entry types

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

7.7.1 Article

```
FUNCTION {article}
{ output.bibitem
  format.authors "author" output.check
  author format.key output
  format.date "year" output.check
  date.block
  format.title "title" output.check
  print.title.addenda
  new.block
  crossref missing$
    { journal "t" change.case$
      "journal" bibinfo.check
      emphasize
      "journal" output.check
```

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

```
      is.online
        { continue.clause
          bbl.online output
          #1 'online.shown :=
        }
        'skip$
      if$
```

```
      format.vol.num.pages output
    }
    { format.article.crossref output.nonnull
      format.pages output
    }
  if$
  new.block
  format.note output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.2 Book

```
FUNCTION {book}
{ output.bibitem
  author empty$
    { editor empty$
      type$ "reference" =
      or
        { type$ "reference" =
            { format.title "title" output.check }
            { format.btitle "title" output.check }
          if$
          print.labeltitle.addenda
          format.date "year" output.check
          year empty$ 'skip$ { date.block } if$
          print.title.addenda
        }
        { format.editors "author and editor" output.check
          format.date "year" output.check
          date.block
          format.btitle "title" output.check
          print.title.addenda
        }
      if$
    }
    { format.authors output.nonnull
      crossref missing$
        { "author and editor" editor either.or.check }
        'skip$
      if$
      format.date "year" output.check
      date.block
      format.btitle "title" output.check
      print.title.addenda
    }
  if$
  format.bvolume output
  format.number.series output
  new.block
  format.edition output
  new.block
  format.note output
  crossref missing$
    { keywords field.or.null "uksi" =
        { mid.sentence 'output.state := }
        { new.sentence }
      if$
      format.publisher.address output
    }
    { new.block
      format.book.crossref output.nonnull
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.3 Unpublished or self-published monograph

```
FUNCTION {booklet}
{ output.bibitem
  author empty$
    { format.avtitle "title" output.check
      print.labeltitle.addenda
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.date "year" output.check
      date.block
      format.avtitle "title" output.check
      print.title.addenda
    }
  if$
  new.block
  format.series.episode output
```

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

```
  is.online
    { continue.clause
      bbl.online output
      #1 'online.shown :=
    }
    'skip$
  if$
```

```
  new.block
  type "type" bibinfo.check output
  new.block
  howpublished "howpublished" bibinfo.check output
  new.block
  format.note output
  publisher empty$ 'skip$
    { new.block}
  if$
  format.publisher.address output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.4 Work in book

```
FUNCTION {inbook}
{ output.bibitem
  author empty$
    { format.editors "author and editor" output.check
      editor format.key output
    }
    { format.authors output.nonnull
      crossref missing$
        { "author and editor" editor either.or.check }
        'skip$
      if$
    }
  if$
  format.date "year" output.check
  date.block
  format.btitle "title" output.check
  print.title.addenda
  new.block
  format.edition output
  new.block
  format.note output
  new.block
  crossref missing$
    { format.publisher.address output
      format.bvolume output
      format.chapter.pages "chapter and pages" output.check
      new.block
      format.number.series output
      new.sentence
    }
    { format.chapter.pages "chapter and pages" output.check
      new.block
      format.book.crossref output.nonnull
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.5 Work in collection

```
FUNCTION {incollection}
{ output.bibitem
  author empty$
    { type$ "inreference" =
        { format.label.booktitle output}
        { format.title "title" output.check
          print.labeltitle.addenda
        }
      if$
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      type$ "inreference" =
        'skip$
        { print.title.addenda }
      if$
    }
    { format.authors output.nonnull
      format.date "year" output.check
      date.block
      format.title "title" output.check
      print.title.addenda
    }
  if$
  new.block
  crossref missing$
    { type$ "inreference" =
      author empty$
      and
        { format.byeditors output }
        { format.in.ed.booktitle "booktitle" output.check }
      if$
      new.block
      format.edition output
      new.block
      type$ "inreference" =
      author empty$
      and
        { format.btitle "title" output.check
          print.title.addenda
          new.block
        }
        'skip$
      if$
      format.note output
      new.block
      format.publisher.address output
      format.bvolume output
      format.number.series output
      format.chapter.pages output
    }
    { format.note output
      new.block
      format.incoll.inproc.crossref output.nonnull
      format.chapter.pages output
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.6 Conference paper

```
FUNCTION {inproceedings}
{ output.bibitem
  format.authors "author" output.check
  author format.key output
  format.date "year" output.check
  date.block
  format.title "title" output.check
  print.title.addenda
  new.block
  crossref missing$
    { format.in.ed.booktitle "booktitle" output.check
      eventyear output
      venue output
      format.note output
      new.sentence
      publisher empty$
        { format.organization.address output }
        { organization "organization" bibinfo.check output
          format.publisher.address output
        }
      if$
      format.bvolume output
      format.number.series output
      format.pages output
    }
    { format.note output
      new.block
      format.incoll.inproc.crossref output.nonnull
      format.pages output
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
FUNCTION {conference} { inproceedings }
```

7.7.7 Legal case studies

```
FUNCTION {jurisdiction}
{ output.bibitem
  author empty$
    { format.btitle "title" output.check
      continue.clause
      format.case.number output
      number empty$
      casenumber empty$
      and
        'new.sentence
        'continue.clause
      if$
      print.labeltitle.addenda
      format.jur.date "year" output.check
      year empty$ 'skip$ { eu.case.check } if$
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.jur.date "year" output.check
      date.block
      continue.clause
      format.btitle "title" output.check
      print.labeltitle.addenda
      print.title.addenda
    }
  if$
  eu.case.check
  format.note output
  note empty$
    'eu.case.check
    'new.block
  if$
  crossref missing$
    { continue.clause
      format.journal.series.vol.pages output
    }
    { format.article.crossref output.nonnull
      format.pages output
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.8 Legislation

```
FUNCTION {uklegislation}
{ output.bibitem
  author empty$
    { format.btitle "title" output.check
      print.labeltitle.addenda
      continue.clause
      format.date emphasize "year" output.check
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.date "year" output.check
      date.block
```

```
   continue.clause
   format.btitle "title" output.check
   print.labeltitle.addenda
   print.title.addenda
  }
 if$
 entrysubtype field.or.null "secondary" =
  { number "number" bibinfo.check output }
  { series empty$ type empty$ and
    { number empty$
      { format.chapter output }
      { continue.clause
       format.series.number.chapter output
      }
     if$
    }
    { chapter empty$
      { new.block }
      { continue.clause }
     if$
     format.series.number.chapter output
    }
   if$
  }
 if$
 note empty$
  'skip$
  { new.block
   format.note output
   new.block
  }
 if$
 publisher empty$ 'skip$
  { entrysubtype field.or.null "secondary" =
    'skip$
    { new.block }
   if$
   format.publisher.address output
  }
 if$
 new.block
 format.eprint output
 format.url output
 fin.entry
}
FUNCTION {eulegislation}
{ output.bibitem
 author empty$
 { format.title "title" output.check
  print.labeltitle.addenda
  continue.clause
  format.eu.date "year" output.check
  print.title.addenda
 }
 { format.authors output
  author format.key output
  format.eu.date "year" output.check
  date.block
  continue.clause
  format.title "title" output.check
  print.labeltitle.addenda
  print.title.addenda
 }
 if$
 eu.case.check
 format.note output
 note empty$
  'eu.case.check
  'new.block
 if$
 crossref missing$
  { continue.clause
   format.journal.series.vol.pages output
```

```
    }
    { format.article.crossref output.nonnull
      format.pages output
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
FUNCTION {legislation}
{ journal empty$
    'uklegislation
    'eulegislation
  if$
}
```

7.7.9 Manual

```
FUNCTION {manual}
{ output.bibitem
  author empty$
    { format.btitle "title" output.check
      print.labeltitle.addenda
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.date "year" output.check
      date.block
      format.btitle "title" output.check
      print.title.addenda
    }
  if$
  month "month" bibinfo.check output
  format.edition output
  new.block
  format.manual.number output
  new.block
  format.note output
  new.block
  format.library output
  organization address new.block.checkb
  format.organization.address output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.10 Image

```
FUNCTION {image}
{ output.bibitem
  author empty$
  { format.btitle "title" output.check
    print.labeltitle.addenda
    format.date "year" output.check
    year empty$ 'skip$ { date.block } if$
    print.title.addenda
  }
  { format.authors output
    author format.key output
    format.date "year" output.check
    date.block
    format.btitle "title" output.check
    print.title.addenda
  }
  if$
  format.edition output
  new.block
  format.note output
  new.block
  format.manual.number output
  new.block
  format.pub.org.lib.address output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.11 Master's thesis

```
FUNCTION {mastersthesis}
{ output.bibitem
  format.authors "author" output.check
  author format.key output
  format.date "year" output.check
  date.block
  format.btitle "title" output.check
  print.title.addenda
  new.block
  format.note output
  new.block
  bbl.mthesis format.thesis.type output.nonnull
  new.block
  school "school" bibinfo.warn output
  address "address" bibinfo.check output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.12 Miscellaneous

```
FUNCTION {misc}
{ output.bibitem
  author empty$
    { format.title "title" output.check
      print.labeltitle.addenda
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.date "year" output.check
      date.block
      format.title "title" output.check
      print.title.addenda
    }
  if$
  new.block
  howpublished "howpublished" bibinfo.check output
  new.block
  format.note output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.13 Doctoral thesis

```
FUNCTION {phdthesis}
{ output.bibitem
  format.authors "author" output.check
  author format.key output
  format.date "year" output.check
  date.block
  format.btitle "title" output.check
  print.title.addenda
  new.block
  format.note output
  new.block
  bbl.phdthesis format.thesis.type output.nonnull
  new.block
  school "school" bibinfo.warn output
  address "address" bibinfo.check output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.14 Conference proceedings

```
FUNCTION {proceedings}
{ output.bibitem
  editor empty$
    { format.btitle "title" output.check
      print.labeltitle.addenda
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      print.title.addenda
    }
    { format.editors output
      editor format.key output
      format.date "year" output.check
      date.block
      format.btitle "title" output.check
      print.title.addenda
    }
  if$
  format.bvolume output
  format.number.series output
  format.note output
  new.sentence
  publisher empty$
    { format.organization.address output }
    { organization "organization" bibinfo.check output
      format.publisher.address output
    }
  if$
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.15 Report

```
FUNCTION {techreport}
{ output.bibitem
  author empty$
    { format.btitle "title" output.check
      print.labeltitle.addenda
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.date "year" output.check
      date.block
      format.btitle "title" output.check
      print.title.addenda
    }
  if$
  new.block
  format.tr.number output
  new.block
```

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

```
  format.note output
  new.block
```

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

```
  library empty$
    'skip$
    { format.note output
      new.block
    }
  if$
```

This bit is the same.

```
  publisher empty$
    { format.institution.address output }
    { institution "institution" bibinfo.check output
      format.publisher.address output
    }
  if$
  new.block
```

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

```
  format.library output
```

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

```
  library empty$
    { format.note output }
    { format.library output }
  if$
```

The rest is the same.

```
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.16 Unpublished works

```
FUNCTION {unpublished}
{ output.bibitem
  author empty$
    { booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
      print.labeltitle.addenda
      format.date "year" output.check
      year empty$ 'skip$ { date.block } if$
      print.title.addenda
    }
    { format.authors output
      author format.key output
      format.date "year" output.check
      date.block
      booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
      print.title.addenda
    }
  if$
  new.block
  format.in.ed.booktitle output
  new.block
  howpublished "howpublished" bibinfo.check output
  new.block
  format.note output
  new.block
  bbl.unpublished output
  new.block
  format.eprint output
  format.url output
  fin.entry
}
```

7.7.17 Aliases and fallbacks

Here are a list of aliases supported by this style.

```
FUNCTION {letter}
{ journal empty$
    'unpublished
    'article
  if$
}
FUNCTION {collection} { book }
FUNCTION {reference} { book }
FUNCTION {inreference} { incollection }
FUNCTION {audio} { booklet }
FUNCTION {movie} { booklet }
FUNCTION {music} { booklet }
FUNCTION {video} { booklet }
FUNCTION {dataset} { manual }
FUNCTION {electronic} { manual }
FUNCTION {online} { manual }
FUNCTION {patent} { manual }
FUNCTION {software} { manual }
FUNCTION {standard} { manual }
FUNCTION {www} { manual }
FUNCTION {thesis} { phdthesis }
FUNCTION {report} { techreport }
```

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

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

7.8 Writing out the bibliography to a file

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

```
READ
```

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

```
FUNCTION {sortify}
{ purify$
  "l" change.case$
}
```

`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*.

```
INTEGERS { len }
FUNCTION {chop.word}
{ 's :=
  'len :=
  s #1 len substring$ =
    { s len #1 + global.max$ substring$ }
    's
  if$
}
```

`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.’

```
FUNCTION {format.lab.names}
{'s :=
 "" 't :=
  #1 'nameptr :=
  s num.names$ 'numnames :=
  numnames 'namesleft :=
    { namesleft #0 > }
    { s nameptr
      "{vv~}{ll}" format.name$
      't :=
      nameptr #1 >
        { nameptr #2 =
          numnames #3 > and
            { "others" 't :=
              #1 'namesleft := }
            'skip$
          if$
          namesleft #1 >
            { ", " * t * }
            { s nameptr "{ll}" format.name$ duplicate$ "others" =
                { 't := }
                { pop$ }
              if$
              t "others" =
                { " " * bbl.etal *
                }
                { bbl.and
                  space.word * t *
                }
              if$
            }
          if$
        }
        't
      if$
      nameptr #1 + 'nameptr :=
      namesleft #1 - 'namesleft :=
    }
  while$
}
```

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

```
FUNCTION {author.editor.key.label}
{ author empty$
    { editor empty$
        { key empty$
            { title empty$
                { cite$ #1 #3 substring$ }
                { title
                  type$ "book" =
                    { emphasize }
                    'skip$
                  if$
                }
              if$
            }
            'key
          if$
        }
        { editor format.lab.names }
      if$
    }
    { author format.lab.names }
  if$
}
```

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

```
FUNCTION {editor.key.label}
{ editor empty$
    { key empty$
        { title empty$
            { cite$ #1 #3 substring$ }
            { title emphasize }
          if$
        }
        'key
      if$
    }
    { editor format.lab.names }
  if$
}
```

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

```
FUNCTION {author.key.label}
{ author empty$
    { key empty$
        { type$ "inreference" =
          booktitle empty$ not
          and
            { booktitle }
            { title empty$
                { cite$ #1 #3 substring$ }
                { title
                  type$ "article" =
                  type$ "incollection" =
                  or
                  type$ "inproceedings" =
                  or
                    type$ "reference" =
                    author empty$
                    and
                  or
                  type$ "misc" =
                  or
                    'skip$
                    { emphasize }
                  if$
                }
              if$
            }
          if$
        }
        'key
      if$
    }
    { author format.lab.names }
  if$
}
```

`calc.short.authors` chooses between the above.

```
FUNCTION {calc.short.authors}
{ type$ "book" =
  type$ "inbook" =
  or
    'author.editor.key.label
    { type$ "proceedings" =
        'editor.key.label
        'author.key.label
      if$
    }
  if$
  'short.list :=
}
```

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

```
FUNCTION {calc.label}
{ calc.short.authors
  short.list
  "("
  *
  year nodate.check duplicate$ empty$
     { pop$ "" }
     'skip$
  if$
  *
  'label :=
}
```

`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.’

```
FUNCTION {sort.format.names}
{ 's :=
  #1 'nameptr :=
  ""
  s num.names$ 'numnames :=
  numnames 'namesleft :=
    { namesleft #0 > }
    { s nameptr
      "{ll{ }}{  f{ }}{  jj{ }}{ vv{}}"
      format.name$ 't :=
      nameptr #1 >
        { "   "  *
          namesleft #1 =
          t "others" =
          and
            { "zzzzz" 't := }
            'skip$
          if$
          t sortify *
        }
        { t sortify * }
      if$
      nameptr #1 + 'nameptr :=
      namesleft #1 - 'namesleft :=
    }
  while$
}
```

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

```
FUNCTION {sort.format.title}
{ 't :=
  "A " #2
    "An " #3
      "The " #4 t chop.word
    chop.word
  chop.word
  sortify
  #1 global.max$ substring$
}
```

`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.

```
FUNCTION {author.editor.sort}
{ author empty$
    { editor empty$
        { title empty$
            { key empty$
                { "to sort, need author, editor, title, or key in " cite$ * warning$
                  ""
                }
                { key sortify }
              if$
            }
            { title sort.format.title }
          if$
        }
        { editor sort.format.names }
      if$
    }
    { author sort.format.names }
  if$
}
```

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

```
FUNCTION {editor.sort}
{ editor empty$
    { title empty$
        { key empty$
            { "to sort, need editor, title, or key in " cite$ * warning$
              ""
            }
            { key sortify }
          if$
        }
        { title sort.format.title }
      if$
    }
    { editor sort.format.names }
  if$
}
```

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

```
FUNCTION {author.sort}
{ author empty$
    { type$ "inreference" =
      booktitle empty$ not
      and
        { booktitle sort.format.title }
        { title empty$
            { key empty$
                { "to sort, need author, title, or key in " cite$ * warning$
                  ""
                }
                { key sortify }
              if$
            }
            { title sort.format.title }
          if$
        }
      if$
    }
    { author sort.format.names }
  if$
}
```

`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.

```
FUNCTION {presort}
{ calc.label
  label sortify
  "    "
  *
  type$ "book" =
  type$ "inbook" =
  or
    'author.editor.sort
    { type$ "proceedings" =
        'editor.sort
        'author.sort
      if$
    }
  if$
  #1 entry.max$ substring$
  'sort.label :=
  sort.label
  *
  "    "
  *
  title field.or.null
  sort.format.title
  *
  #1 entry.max$ substring$
  'sort.key$ :=
}
```

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

```
ITERATE {presort}
SORT
```

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

```
STRINGS { last.label next.extra }
INTEGERS { last.extra.num last.extra.num.extended last.extra.num.blank number.label }
FUNCTION {initialize.extra.label.stuff}
{ #0 int.to.chr$ 'last.label :=
  "" 'next.extra :=
  #0 'last.extra.num :=
  "a" chr.to.int$ #1 - 'last.extra.num.blank :=
  last.extra.num.blank 'last.extra.num.extended :=
  #0 'number.label :=
}
FUNCTION {forward.pass}
{ last.label label =
    { last.extra.num #1 + 'last.extra.num :=
      last.extra.num "z" chr.to.int$ >
       { "a" chr.to.int$ 'last.extra.num :=
         last.extra.num.extended #1 + 'last.extra.num.extended :=
       }
       'skip$
      if$
      last.extra.num.extended last.extra.num.blank >
        { last.extra.num.extended int.to.chr$
          last.extra.num int.to.chr$
          * 'extra.label := }
        { last.extra.num int.to.chr$ 'extra.label := }
      if$
    }
    { "a" chr.to.int$ 'last.extra.num :=
      "" 'extra.label :=
      label 'last.label :=
    }
  if$
  number.label #1 + 'number.label :=
}
FUNCTION {reverse.pass}
{ next.extra "b" =
    { "a" 'extra.label := }
    'skip$
  if$
  extra.label 'next.extra :=
  extra.label
  duplicate$ empty$
```

```
    'skip$
    { "{\natexlab{" swap$ * "}}" * }
  if$
  'extra.label :=
  label extra.label * 'label :=
}
```

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.

```
EXECUTE {initialize.extra.label.stuff}
ITERATE {forward.pass}
REVERSE {reverse.pass}
FUNCTION {bib.sort.order}
{ sort.label
  "    "
  *
  year field.or.null sortify
  *
  "    "
  *
  title field.or.null
  sort.format.title
  *
  #1 entry.max$ substring$
  'sort.key$ :=
}
ITERATE {bib.sort.order}
SORT
```

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

```
FUNCTION {begin.bib}
{ preamble$ empty$
    'skip$
    { preamble$ write$ newline$ }
  if$
  "\begin{thebibliography}{" number.label int.to.str$ * "}" *
  write$ newline$
  "\providecommand{\natexlab}[1]{#1}"
  write$ newline$
  "\providecommand{\url}[1]{\texttt{#1}}"
  write$ newline$
  "\providecommand{\urlprefix}{Available from: }"
  write$ newline$
  "\providecommand{\urldateprefix}{Accessed }"
  write$ newline$
  "\providecommand{\selectlanguage}[1]{\relax}"
  write$ newline$
  "\providecommand{\bibinfo}[2]{#2}"
  write$ newline$
  "\providecommand{\eprint}[2][]{\url{#2}}"
  write$ newline$
}
EXECUTE {begin.bib}
EXECUTE {init.state.consts}
```

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

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

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

```
FUNCTION {end.bib}
{ newline$
  "\end{thebibliography}" write$ newline$
}
EXECUTE {end.bib}
```