

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

Maintainer: Alex Ball*

Package v3.2 – 16 January 2020

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](#) and [ipaex](#) packages installed; this is just for the documentation, not for the BibTeX style itself. To generate the version 1 tests and documentation, compile `bath-bst-v1.tex` with LuaLaTeX and BibTeX.
3. If you are using TeX Live, find your home TeX tree using the following command at the command prompt/terminal:

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

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

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

2 Choosing which variant to use

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

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

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

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

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

One other change worth noting regards URL access dates. In version 1, the advice was to put them in `urldate`. With version 2 the advice is now 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}
\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{williams1997edd}`, 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 (Williams, 1997), 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!

3.1 Migrating from version 2 to version 3

If you have previously used version 2 of the style, you may need to update your .bib file to accommodate the following changes in 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 Examples

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

4.1 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


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


Electronic book


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

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

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

RX


 Hodds, J., 2016. *Referencing ebooks* [Kindle version 4.18]. Bath: University of Bath.


 Hodds, J., 2016. *Referencing ebooks* [Kindle version 4.18]. Bath: University of Bath.

```
@book{hodds2016re,  
  author = {Hodds, J.},  
  year = {2016},  
  title = {Referencing ebooks},  
  titleaddon = {Kindle version 4.18},  
  address = {Bath},  
  publisher = {University of Bath}}
```

RX

One 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

4.2 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

Electronic journal article

📄 Williams, F., 1997. Electronic document delivery: a trial in an academic library. *Ariadne* [Online], 10. Available from: <http://www.ariadne.ac.uk/issue10/edd/> [Accessed 5 December 1997].

⚙️ Williams, F., 1997. Electronic document delivery: a trial in an academic library. *Ariadne* [Online], 10. Available from: <http://www.ariadne.ac.uk/issue10/edd/> [Accessed 5 December 1997].

```
@article{williams1997edd,  
  author = {Williams, F.},  
  year = {1997},  
  title = {Electronic Document Delivery: a Trial in an Academic Library},  
  journal = {Ariadne},  
  volume = {10},  
  url = {http://www.ariadne.ac.uk/issue10/edd/},  
  urlyear = {5 December 1997}}
```

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 in a digital repository

💡 The Harvard (Bath) style gives you a choice whether to treat preprints as pre-publication articles (above) or as grey literature (below). In the latter case, the `techreport` entry type is the best match semantically.

📖 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].

```
@techreport{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

Newspaper article

🔧 Give the issue's day and month as the 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}},  
  volume = {29 July},  
  pages = {16c}}
```

RX

📖 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}},  
  volume = {4 June},  
  pages = {28a}}
```

RX

4.3 Conference papers

❗ As in the standard BibTeX styles, `conference` is a legacy (and highly deprecated) alias for `inproceedings`.



Conference paper (when proceedings have a named editor)

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

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

RX

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



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

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

RX


4.4 Grey literature

Thesis/dissertation


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


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

RX

 Internally, `thesis` is an alias for `phdthesis`.


Report


 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.

```
@techreport{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

 Internally, `report` is an alias for `techreport`.


Standard


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

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

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


RX

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

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

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

RX

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

Patent

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

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

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

RX

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

Unpublished written material and personal communications

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

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

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

RX

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

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

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

RX

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


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


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

RX

4.5 Audiovisual materials


Image


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


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

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

RI


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


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

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

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


RI

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

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

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

RI

 Use the **library** field to record the archive and register number. If you also wish to provide the name of the organisation that published the image, use the **publisher** field if the location you provide relates to the publisher, but the **organization** field if the location relates to the archive.

Map

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

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

```
@manual{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

💡 The book entry type would also work for this reference.

Film, video or DVD

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

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

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

RX

📘 Internally, movie and video are aliases for booklet.

Streamed video (YouTube, TED Talks, etc.)

📖 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

📖 University of Bath, 2015. *What is engineering?* [Online]. Available from: <https://www.youtube.com/watch?v=NoyZarq-Zbo> [Accessed 12 January 2016].

⚙️ University of Bath, 2015. *What is engineering?* [Online]. Available from: <https://www.youtube.com/watch?v=NoyZarq-Zbo> [Accessed 12 January 2016].

```
@video{uob2015wie,  
  author = {{University of Bath}},  
  year = {2015},  
  title = {What is Engineering?},  
  url = {https://www.youtube.com/watch?v=NoyZarq-Zbo},  
  urlyear = {12 January 2016}}
```

RI

📖 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}}
```

RJ

Television or radio broadcast

📺 *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

💡 Use type for the medium and note for the channel, date and time.

📌 Internally, audio and music are aliases for booklet.

Music score

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

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

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

RX

4.6 Digital media

Website/webpage

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

📖 Holland, M., 2002. *Guide to citing internet sources* [Online]. Poole: Bournemouth University. Available from: http://www.bournemouth.ac.uk/library/using/guide_to_citing_internet_sourc.html [Accessed 4 November 2002].

⚙️ Holland, M., 2002. *Guide to citing internet sources* [Online]. Poole: Bournemouth University. Available from: http://www.bournemouth.ac.uk/library/using/guide_to_citing_internet_sourc.html [Accessed 4 November 2002].

```
@online{holland2002gci,  
  author = {Holland, M.},  
  year = {2002},  
  title = {Guide to Citing Internet Sources},  
  address = {Poole},  
  organization = {Bournemouth University},  
  url = {http://www.bournemouth.ac.uk/library/using/guide_to_citing_internet_sourc.html},  
  urlyear = {4 November 2002}}
```

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

Dataset

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

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

RX

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

Computer program

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

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

RX

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

4.7 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

📖 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

- 💡 The following 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.

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

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

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

RX

💡 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 vsekidnevnie v Balgariya* [Roma everyday life in Bulgaria]. Veliko Tarnovo: Faber.

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

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

RX

4.8 Legal references: UK legislation and parliamentary reports

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



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

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.

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},
  number = {30}}
```

RX

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},
  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},
  number = {6041}}
```

RX

4.9 Legal references: EU legislation and reports

EU regulation or directive, decision, recommendation or opinion

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

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

RX

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

<pre>\defcitealias{eu.dir1984/2003}{% Council Regulation [EC] 1984/2003} \citealias{eu.dir1984/2003}</pre>	<pre>(Council Regulation [EC] 1984/2003)</pre>
--	--

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.

```
@techreport{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

4.10 Legal references: case reports

Legal case study

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

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

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

RX

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

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

References

- Adams, G., 2009. *Test citation one*. London: Imperial College Bookstall.
- Adams, G., 2014. *Test citation two*. Oxford: Oxford University Press.
- Adams, G., 2017. *Test citation three*. Cambridge: Cambridge University Press.
- Alessandrini Srl and others v. Commission* (C-295/03 P) [2005] ECR I-5700.
- Andrews, J. and Dury, A., 1773. *Map of Wiltshire*. 1 inch to 2 miles. Devizes: Wiltshire Record Society.
- Anon., 1946. *Prototype electric bicycle displayed at the ‘Britain Can Make It’ exhibition organised by the Council of Industrial Design and held at the Victoria and Albert Museum, London, 1946. Designed by B. G. Bowden* [Photograph]. At: London. Design Council Slide Collection.
- The Archers*, 2006. Radio. BBC Radio 4, 23 August. 19.02 hrs.
- Aristotle, 2007. *Nicomachean ethics* (W.D. Ross, Trans.). South Dakota: NuVisions.
- ASTM, 2019. *ASTM D1655 - 19 Standard specification for aviation turbine fuels*. West Conshohocken, Pa.: ASTM.
- Beethoven, L. van, 1950. *Symphony no.1 in C, Op.21*. Harmondsworth: Penguin.
- BRE, 2007. *Designing quality buildings: a BRE guide*. (Report 497). Bracknell: BRE.
- 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].
- Council Regulation (EC) 1831/2003 of 8 April 2003 introducing a system for the statistical monitoring of trade in bluefin tuna, swordfish and big eye tuna within the Community [2003] *OJ* L295.
- Crawford, G.I., 1965. Oxygen in metals. In: J.M.A. Lenihan and S.J. Thompson, eds. *Activation analysis: Proceedings of a NATO Advanced Study Institute, 2–4 August 1964*, Glasgow. London: Academic Press, pp.113–118.
- 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.
- Great Britain. Ministry of Defence, 2004. *Delivering security in a changing world: defence white paper*. (Cm. 6041). London: TSO.
- Great Britain. Parliament. House of Commons, 1988. *Local government finance bill*. (Bills | 1987/88, 66). London: HMSO.
- Great Britain. Parliament. House of Commons, 2004. *National Savings investment deposits: account 2002–2003*. (HC 2003/04, 30). London: National Audit Office.
- Great Britain. Parliament. House of Lords, 1987. *Social fund (maternity and funeral expenses) bill*. (HL 1986/87, (66)). London: HMSO.
- Hadley, S., 2015. *Biomechanics: introductory reading, BM289: sport biomechanics*. University of Bath. Unpublished.
- Harris, G., 2013. *Focus group recommendations: internal task group report*. Unpublished.
- Haurant, S., 2004. Britain's borrowing hits £1 trillion. *The Guardian*, 29 July, p.16c.
- Haynes, W.M., ed., 2014. *CRC handbook of chemistry and physics* [Online]. 94th ed. Boca Raton, Fla.: CRC Press/Taylor and Francis. Available from: <http://www.hbcnetbase.com> [Accessed 16 June 2016].
- Hodds, J., 2016. *Referencing ebooks* [Kindle version 4.18]. Bath: University of Bath.
- Holland, M., 2002. *Guide to citing internet sources* [Online]. Poole: Bournemouth University. Available from: http://www.bournemouth.ac.uk/library/using/guide_to_citing_internet_sourc.html [Accessed 4 November 2002].
- Hua, L., 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue*, 1, pp.168–79.
- Hua, L. 華林甫, 1999. Qingdai yilai Sanxia diqu shuihan zaihai de chubu yanjiu 清代以來三峽地區水旱災害的初步研 [A preliminary study of floods and droughts in the Three Gorges region since the Qing dynasty]. *Zhongguo shehui kexue* 中國社會科學, 1, pp.168–79.
- The Human Medicines Regulations 2012* [Online], No.1916, United Kingdom: HMSO. Available from: http://www.legislation.gov.uk/ukxi/2012/1916/pdfs/ukxi_20121916_en.pdf [Accessed 17 April 2016].
- 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].
- 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.
- 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.
- 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].
- 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].
- 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.
- 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, 2015. *What is engineering?* [Online]. Available from: <https://www.youtube.com/watch?v=NoyZarq-Zbo> [Accessed 12 January 2016].
- Williams, F., 1997. Electronic document delivery: a trial in an academic library. *Ariadne* [Online], 10. Available from: <http://www.ariadne.ac.uk/issue10/edd/> [Accessed 5 December 1997].

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).

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](#).

5 Implementation

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

5.1 Data model

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

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

5.2 Writing references out to a file

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

```

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

```

We also define two variables to store strings in.

```

71 STRINGS { s t }

```

The `output.nonnull` function removes an element from the stack, writes the appropriate delimiter to the `.bbl` file depending on the state, then restores the original element. Afterwards, the output state is set to `mid.sentence`.

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

```

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

```

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’.

```

96 FUNCTION {end.clause}
97 { output.state mid.clause =
98   { mid.sentence 'output.state := }
99   'skip$
100  if$
101 }

```

Before writing anything out, we need to make sure there is something to be printed. We provide two functions for this: `output` is silent, while `output.check` will write a warning to the log that the field is empty. The latter consumes an additional string (at the top of the stack) for printing in the warning message.

```

102 FUNCTION {output}
103 { duplicate$ empty$
104   { pop$ end.clause }
105   'output.nonnull

```

```

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

```

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`.

```

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

```

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

```

147 FUNCTION {date.block}
148 { new.block
149 }

```

The `bibinfo` functions consume two tokens: the value and name of a field respectively. If the value is non-null, it is written out to the `.bbl` file in the form `\bibinfo{field name}{field value}`. If the value is missing, the tokens are replaced with an empty string; if the value is empty, it is put back on the stack. The `check` version does this silently, while the `warn` version issues warnings in the log file if the value is missing or empty.

```

150 FUNCTION {bibinfo.check}
151 { swap$
152   duplicate$ missing$
153   { pop$ pop$
154     ""
155   }
156   { duplicate$ empty$
157     { swap$ pop$
158       }
159     { swap$
160       "\bibinfo{" swap$ * "}{ " * swap$ * "}" *
161     }
162     if$
163   }

```

```

164   if$
165 }
166 FUNCTION {bibinfo.warn}
167 { swap$
168   duplicate$ missing$
169   { swap$ "missing " swap$ * " in " * cite$ * warning$ pop$
170     ""
171   }
172   { duplicate$ empty$
173     { swap$ "empty " swap$ * " in " * cite$ * warning$
174       }
175     { swap$
176       "\bibinfo{" swap$ * "}" * swap$ * "}" *
177     }
178   if$
179 }
180 if$
181 }

```

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.

```

182 FUNCTION {fin.entry}
183 { add.period$
184   write$
185   newline$
186 }

```

5.3 Convenience functions

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

```

188 FUNCTION {not}
189 { { #0 }
190   { #1 }
191   if$
192 }
193 FUNCTION {and}
194 { 'skip$
195   { pop$ #0 }
196   if$
197 }
198 FUNCTION {or}
199 { { pop$ #1 }
200   'skip$
201   if$
202 }

```

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

```

203 FUNCTION {new.block.checkb}
204 { empty$
205   swap$ empty$
206   and
207   'skip$
208   'new.block
209   if$
210 }

```

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

```

211 FUNCTION {field.or.null}
212 { duplicate$ empty$
213   { pop$ "" }
214   'skip$
215   if$
216 }

```

The `emphasize` function applies emphasis.

```

217 FUNCTION {emphasize}
218 { duplicate$ empty$

```

```

219     { pop$ "" }
220     { "\emph{" swap$ * "}" * }
221     if$
222 }

```

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.

```

223 FUNCTION {tie.or.space.prefix}
224 { duplicate$ text.length$ #3 <
225   { "~" }
226   { " " }
227   if$
228   swap$
229 }

```

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

```

230 FUNCTION {capitalize}
231 { "u" change.case$ "t" change.case$ }

```

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

```

232 FUNCTION {space.word}
233 { " " swap$ * " " * }

```

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

```

234 FUNCTION {either.or.check}
235 { empty$
236   'pop$
237   { "can't use both " swap$ * " fields in " * cite$ * warning$ }
238   if$
239 }

```

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

```

240 FUNCTION {is.online}
241 { type$ "software" =
242   { #0 }
243   { url empty$ doi empty$ and
244     { type$ "online" =
245       { #1 }
246       { #0 }
247       if$
248     }
249     { #1 }
250     if$
251   }
252   if$
253 }

```

5.4 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.

```

254 FUNCTION {bbl.and}
255 { "and" }
256
257 FUNCTION {bbl.etal}
258 { "et-al." }
259
260 FUNCTION {bbl.editors}
261 { "eds" }
262
263 FUNCTION {bbl.editor}
264 { "ed." }
265
266 FUNCTION {bbl.edby}

```

```

267 { "edited by" }
268
269 FUNCTION {bb1.translator}
270 { "Trans." }
271
272 FUNCTION {bb1.edition}
273 { "ed." }
274
275 FUNCTION {bb1.volume}
276 { "vol." }
277
278 FUNCTION {bb1.of}
279 { "of" }
280
281 FUNCTION {bb1.number}
282 { "no." }
283
284 FUNCTION {bb1.nr}
285 { "no." }
286
287 FUNCTION {bb1.in}
288 { "in" }
289
290 FUNCTION {bb1.pages}
291 { "pp." }
292
293 FUNCTION {bb1.page}
294 { "p." }
295
296 FUNCTION {bb1.chapter}
297 { "chap." }
298
299 FUNCTION {bb1.techrep}
300 { "Tech. Rep." }
301
302 FUNCTION {bb1.mthesis}
303 { "Master's thesis" }
304
305 FUNCTION {bb1.phdthesis}
306 { "Ph.D. thesis" }
307
308 FUNCTION {bb1.first}
309 { "1st" }
310
311 FUNCTION {bb1.second}
312 { "2nd" }
313
314 FUNCTION {bb1.third}
315 { "3rd" }
316
317 FUNCTION {bb1.fourth}
318 { "4th" }
319
320 FUNCTION {bb1.fifth}
321 { "5th" }
322
323 FUNCTION {bb1.st}
324 { "st" }
325
326 FUNCTION {bb1.nd}
327 { "nd" }
328
329 FUNCTION {bb1.rd}
330 { "rd" }
331
332 FUNCTION {bb1.th}
333 { "th" }
334
335 FUNCTION {bb1.commission}
336 { "Commission Decision" }
337
338 FUNCTION {bb1.online}
339 { "[Online]" }

```

```

340
341 FUNCTION {bbl.nodate}
342 { "n.d." }
343
344 FUNCTION {bbl.inpreparation}
345 { "preprint" }
346
347 FUNCTION {bbl.submitted}
348 { "preprint" }
349
350 FUNCTION {bbl.inpress}
351 { "in press" }
352
353 FUNCTION {bbl.unpublished}
354 { "Unpublished" }

```

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

```

356 MACRO {jan} {"January"}
357
358 MACRO {feb} {"February"}
359
360 MACRO {mar} {"March"}
361
362 MACRO {apr} {"April"}
363
364 MACRO {may} {"May"}
365
366 MACRO {jun} {"June"}
367
368 MACRO {jul} {"July"}
369
370 MACRO {aug} {"August"}
371
372 MACRO {sep} {"September"}
373
374 MACRO {oct} {"October"}
375
376 MACRO {nov} {"November"}
377
378 MACRO {dec} {"December"}

```

This function allows us to convert cardinal numbers to English ordinals.

```

380 FUNCTION {eng.ord}
381 { duplicate$ "1" swap$ *
382   #-2 #1 substring$ "1" =
383   { bbl.th * }
384   { duplicate$ #-1 #1 substring$
385     duplicate$ "1" =
386     { pop$ bbl.st * }
387     { duplicate$ "2" =
388       { pop$ bbl.nd * }
389       { "3" =
390         { bbl.rd * }
391         { bbl.th * }
392         if$
393       }
394     if$
395   }
396   if$
397 }
398 if$
399 }

```

5.5 Name handling

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


```

401 INTEGERS { nameptr namesleft numnames }
402 STRINGS { bibinfo }
403 FUNCTION {format.names}
404 { 'bibinfo :=
405   duplicate$ empty$
406   'skip$
407   { 's :=
408     "" 't :=
409     #1 'nameptr :=
410     s num.names$ 'numnames :=
411     numnames 'namesleft :=
412     { namesleft #0 > }
413     { s nameptr
414       "{ll}{, jj}{, f{.}.}{-vv}"
415       format.name$
416       bibinfo bibinfo.check
417       't :=
418       nameptr #1 >
419       { namesleft #1 >
420         { ", " * t * }
421         { s nameptr "{ll}" format.name$ duplicate$ "others" =
422           { 't := }
423           { pop$ }
424           if$
425           t "others" =
426           { " " * bbl.etal *
427             }
428           { bbl.and
429             space.word * t *
430             }
431           if$
432           }
433           if$
434           }
435           't
436           if$
437           nameptr #1 + 'nameptr :=
438           namesleft #1 - 'namesleft :=
439           }
440           while$
441           }
442           if$
443           }

```

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

```

444 FUNCTION {format.names.ed}
445 { 'bibinfo :=
446   duplicate$ empty$
447   'skip$
448   { 's :=
449     "" 't :=
450     #1 'nameptr :=
451     s num.names$ 'numnames :=
452     numnames 'namesleft :=
453     { namesleft #0 > }
454     { s nameptr
455       "{f{.}.-}{vv-}{ll}{ jj}"
456       format.name$
457       bibinfo bibinfo.check
458       't :=
459       nameptr #1 >
460       { namesleft #1 >
461         { ", " * t * }
462         { s nameptr "{ll}" format.name$ duplicate$ "others" =
463           { 't := }
464           { pop$ }
465           if$
466           t "others" =
467           { " " * bbl.etal *
468             }
469           { bbl.and

```

```

470         space.word * t *
471     }
472     if$
473 }
474 if$
475 }
476 't
477 if$
478 nameptr #1 + 'nameptr :=
479 namesleft #1 - 'namesleft :=
480 }
481 while$
482 }
483 if$
484 }

```

format.authors returns author names.

```

485 FUNCTION {format.authors}
486 { author "author" format.names
487 }

```

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

```

488 FUNCTION {get.bbl.editor}
489 { editor num.names$ #1 > 'bbl.editors 'bbl.editor if$ }
490 FUNCTION {format.editors}
491 { editor "editor" format.names duplicate$ empty$ 'skip$
492   { ", " *
493     " " *
494     get.bbl.editor
495     *
496   }
497 if$
498 }

```

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

```

499 FUNCTION {format.translators}
500 { translator "translator" format.names.ed duplicate$ empty$ 'skip$
501   { ", " *
502     " " *
503     bbl.translator *
504     "(" swap$ *
505     ")" *
506   }
507 if$
508 }

```

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.

```

509 FUNCTION {format.full.names}
510 { 's :=
511   "" 't :=
512   #1 'nameptr :=
513   s num.names$ 'numnames :=
514   numnames 'namesleft :=
515   { namesleft #0 > }
516   { s nameptr
517     "{vv~}{ll}" format.name$
518     't :=
519     nameptr #1 >
520     { namesleft #1 >
521       { ", " * t * }
522       { s nameptr "{ll}" format.name$ duplicate$ "others" =
523         { 't := }
524         { pop$ }
525         if$
526         t "others" =
527         { " " * bbl.etal *
528         }

```

```

529         { bbl.and
530           space.word * t *
531         }
532         if$
533       }
534       if$
535     }
536     't
537     if$
538     nameptr #1 + 'nameptr :=
539     namesleft #1 - 'namesleft :=
540   }
541   while$
542 }

```

`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.

```

543 FUNCTION {author.editor.key.full}
544 { author empty$
545   { editor empty$
546     { key empty$
547       { cite$ #1 #3 substring$ }
548       'key
549       if$
550     }
551     { editor format.full.names }
552     if$
553   }
554   { author format.full.names }
555   if$
556 }

```

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

```

557 FUNCTION {editor.key.full}
558 { editor empty$
559   { key empty$
560     { cite$ #1 #3 substring$ }
561     'key
562     if$
563   }
564   { editor format.full.names }
565   if$
566 }

```

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

```

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

```

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

```

577 FUNCTION {make.full.names}
578 { type$ "book" =
579   type$ "inbook" =
580   or
581   'author.editor.key.full
582   { type$ "proceedings" =
583     'editor.key.full
584     'author.key.full
585     if$
586   }

```

```
587   if$
588 }
```

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

```
589 FUNCTION {output.bibitem}
590 { newline$
591   "\bibitem[{" write$
592   label write$
593   "]" make.full.names duplicate$ short.list =
594     { pop$ }
595     { * }
596     if$
597     "]"{" * write$
598     cite$ write$
599     "]" write$
600     newline$
601     ""
602     before.all 'output.state :=
603 }
```

`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 a last resort to stop an entry beginning with the date.

```
604 FUNCTION {format.key}
605 { empty$
606   { key field.or.null }
607   { "" }
608   if$
609 }
```

5.6 Titles

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

```
610 FUNCTION {select.language}
611 { duplicate$ empty$
612   'pop$
613   { language empty$
614     'skip$
615     { "\selectlanguage{" language * "}" * swap$ * "}" * }
616     if$
617   }
618   if$
619 }
```

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

```
620 FUNCTION {format.title}
621 { title
622   duplicate$ empty$ 'skip$
623   { "t" change.case$ }
624   if$
625   "title" bibinfo.check
626   duplicate$ empty$
627   'skip$
628   { select.language }
629   if$
630 }
```

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

```

631 FUNCTION {format.btitle}
632 { title
633   duplicate$ empty$ 'skip$
634   { "t" change.case$ }
635   if$
636   "title" bibinfo.check
637   duplicate$ empty$ 'skip$
638   { emphasize
639     select.language
640   }
641   if$
642 }

```

format.titleaddon returns the titleaddon field.

```

643 FUNCTION {format.titleaddon}
644 { titleaddon
645   duplicate$ empty$ 'skip$
646   %<bst1> { "titleaddon" bibinfo.check }
647   %<bst2> { "[" swap$ "titleaddon" bibinfo.check * "]" * }
648   if$
649 }

```

format.version returns the version field.

```

650 FUNCTION {format.version}
651 { version
652   duplicate$ empty$ 'skip$
653   { "(v." swap$ *
654     "version" bibinfo.check
655     ")" *
656   }
657   if$
658 }

```

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

```

659 FUNCTION {get.title.addenda}
660 { continue.clause
661   format.version output
662   continue.clause
663   format.titleaddon output
664   continue.clause
665   %<*bst2>
666   library empty$
667   type$ "image" =
668   or
669   is.online
670   and
671   journal empty$
672   and
673   type empty$
674   type$ "booklet" = not
675   type$ "audio" = not
676   and
677   type$ "video" = not
678   and
679   type$ "music" = not
680   and
681   type$ "movie" = not
682   and
683   or
684   and
685   { bbl.online output }
686   'skip$
687   if$
688   continue.clause
689   %</bst2>
690   format.translators output
691 }

```

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

```

692 FUNCTION {word.in}
693 { bbl.in capitalize
694   ":" *
695   " " * }

```

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

```

696 FUNCTION {format.booktitle}
697 { booktitle
698   duplicate$ empty$ 'skip$
699   { "t" change.case$ }
700   if$
701   "booktitle" bibinfo.check
702   emphasize
703 }

```

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

```

704 FUNCTION {format.in.ed.booktitle}
705 { format.booktitle duplicate$ empty$ 'skip$
706   { editor "editor" format.names.ed duplicate$ empty$ 'pop$
707     { "," *
708       " " *
709       get.bbl.editor add.period$
710       " " *
711       * swap$
712       *
713       word.in swap$ *
714     }
715     if$
716   }
717   if$
718 }

```

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

```

719 FUNCTION {legal.journal.shape}
720 { duplicate$ "OJ" =
721   'emphasize
722   'skip$
723   if$
724 }

```

5.7 Dates

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

```

725 FUNCTION {nodate.check}
726 {
727   %<bst1> skip$
728   %<*bst2>
729   duplicate$ empty$
730   sortyear empty$
731   and
732   { options field.or.null "nodate" =
733     'skip$
734     { pop$ bbl.nodate }
735     if$
736   }
737   'skip$
738   if$
739   %</bst2>
740 }

```

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

```

741 FUNCTION {format.date}
742 { year "year" bibinfo.check nodate.check
743   extra.label *
744 }

```

`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.

```

745 FUNCTION {format.date.brackets}
746 { format.date
747   "[" swap$ * "]" *
748 }
749 FUNCTION {format.date.parens}
750 { format.date
751   "(" swap$ * ")" *
752 }
753 FUNCTION {format.jur.date.essential}
754 { options field.or.null "scottish-style" =
755   keywords field.or.null "sc" =
756   or
757     'format.date
758     'format.date.brackets
759   if$
760 }
761 FUNCTION {format.jur.date}
762 { journal field.or.null 's :=
763   s "OJ" =
764   s "ECR" =
765   or
766     'format.date.brackets
767     { options field.or.null "year-essential=false" =
768       'format.date.parens
769       { volume empty$
770         'format.jur.date.essential
771         { options field.or.null "year-essential=true" =
772           'format.jur.date.essential
773           'format.date.parens
774         if$
775       }
776     if$
777   }
778 if$
779 }
780 if$
781 }

```

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

```

782 FUNCTION {format.eu.date}
783 { journal field.or.null 's :=
784   s "OJ" =
785   s "ECR" =
786   or
787     'format.date.brackets
788     'format.date
789   if$
790 }

```

5.8 Series, volumes, editions

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

```

791 FUNCTION {format.bvolume}
792 { volume empty$
793   { "" }
794   { bbl.volume volume tie.or.space.prefix
795     "volume" bibinfo.check * *
796     series "series" bibinfo.check
797     duplicate$ empty$ 'pop$

```

```

798     { emphasize ", " * swap$ * }
799     if$
800     "volume and number" number either.or.check
801   }
802   if$
803 }

```

`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.

```

804 FUNCTION {format.number.series}
805 { volume empty$
806   { number empty$
807     { series field.or.null }
808     { series empty$
809       { number "number" bibinfo.check }
810       { series "series" bibinfo.check
811         number tie.or.space.prefix "number" bibinfo.check * *
812       }
813       if$
814     }
815     if$
816   }
817   { "" }
818   if$
819 }

```

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

```

820 FUNCTION {is.num}
821 { chr.to.int$
822   duplicate$ "0" chr.to.int$ < not
823   swap$ "9" chr.to.int$ > not and
824 }

```

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

```

825 FUNCTION {extract.num}
826 { duplicate$ 't :=
827   "" 's :=
828   { t empty$ not }
829   { t #1 #1 substring$
830     t #2 global.max$ substring$ 't :=
831     duplicate$ is.num
832     { s swap$ * 's := }
833     { pop$ "" 't := }
834     if$
835   }
836   while$
837   s empty$
838   'skip$
839   { pop$ s }
840   if$
841 }

```

`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.

```

842 FUNCTION {convert.edition}
843 { extract.num "1" change.case$ 's :=
844   s "first" = s "1" = or
845   { bbl.first 't := }
846   { s "second" = s "2" = or
847     { bbl.second 't := }
848     { s "third" = s "3" = or
849       { bbl.third 't := }
850       { s "fourth" = s "4" = or
851         { bbl.fourth 't := }
852         { s "fifth" = s "5" = or
853           { bbl.fifth 't := }

```



```

854         { s #1 #1 substring$ is.num
855           { s eng.ord 't := }
856           { edition 't := }
857             if$
858           }
859         if$
860       }
861     if$
862   }
863   if$
864 }
865 if$
866 }
867 if$
868 t
869 }

```

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

```

870 FUNCTION {format.edition}
871 { edition duplicate$ empty$ 'skip$
872   { convert.edition
873     output.state mid.sentence =
874     { "l" }
875     { "t" }
876     if$ change.case$
877     "edition" bibinfo.check
878     " " * bbl.edition *
879   }
880   if$
881 }

```

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.

```

882 FUNCTION {format.thesis.type}
883 { type duplicate$ empty$
884   'pop$
885   { swap$ pop$
886     "t" change.case$ "type" bibinfo.check
887   }
888   if$
889 }

```

If the type is provided, `format.booklet.type` writes the type field with an initial capital. Then, if this is an online work with no author/editor, the online label is added. If the type is not provided, it returns an empty string.

```

890 FUNCTION {format.booklet.type}
891 { type duplicate$ empty$
892   { pop$ "" }
893   { "type" bibinfo.check
894     is.online author empty$ and editor empty$ and
895     { " " * bbl.online * }
896     'skip$
897     if$
898   }
899   if$
900 }

```

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

If the series begins ‘HL_□’, 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.

```

901 FUNCTION {add.cmd.number}
902 { ". " swap$ "number" bibinfo.check *
903 }

```

```

904 FUNCTION {format.legr.number}
905 { series field.or.null 's :=
906   s "C" =
907     'add.cmd.number
908     { s "Cd" =
909       'add.cmd.number
910       { s "Cmd" =
911         'add.cmd.number
912         { s "Cmnd" =
913           'add.cmd.number
914           { s "Cm" =
915             'add.cmd.number
916             { s #1 #3 substring$ 't :=
917               t "HL " =
918                 { ", (" swap$
919                   "number" bibinfo.check *
920                   ")" *
921                 }
922                 { ", " swap$
923                   "number" bibinfo.check *
924                 }
925               if$
926             }
927           if$
928         }
929       if$
930     }
931   if$
932 }
933 if$
934 }
935 if$
936 }
937 FUNCTION {format.tr.number}
938 { series duplicate$ empty$
939   { pop$
940     type duplicate$ empty$
941     { pop$
942       number duplicate$ empty$
943       { pop$ ""
944       }
945       { "(" swap$
946         "number" bibinfo.check
947         * ")" *
948       }
949     if$
950   }
951   { "(" swap$
952     "t" change.case$ "type" bibinfo.check
953     *
954     number duplicate$ empty$
955     'pop$
956     { tie.or.space.prefix
957       "number" bibinfo.check
958       * *
959     }
960     if$
961     ")" *
962   }
963   if$
964 }
965 { "(" swap$ *
966   type duplicate$ empty$
967   { pop$
968     number duplicate$ empty$
969     'pop$
970     { format.legr.number * }
971   if$
972 }
973 { "t" change.case$ "type" bibinfo.check
974 *
975   number duplicate$ empty$
976   'pop$

```

```

977         { tie.or.space.prefix
978           "number" bibinfo.check
979           * *
980         }
981     if$
982 }
983 if$
984 ")" *
985     continue.clause
986 }
987 if$
988 }

```

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

```

989 FUNCTION {format.manual.number}
990 { series duplicate$ empty$
991   { pop$
992     type duplicate$ empty$
993     { pop$
994       number duplicate$ empty$
995       { pop$ ""
996       }
997       { "number" bibinfo.check
998       }
999     if$
1000   }
1001   { "t" change.case$ "type" bibinfo.check
1002     number duplicate$ empty$
1003     'pop$
1004     { tie.or.space.prefix
1005       "number" bibinfo.check
1006       * *
1007     }
1008     if$
1009   }
1010   if$
1011 }
1012 { number duplicate$ empty$
1013   'pop$
1014   { tie.or.space.prefix
1015     "number" bibinfo.check
1016     * *
1017   }
1018   if$
1019   type duplicate$ empty$
1020   'pop$
1021   { ", " swap$ *
1022     "t" change.case$ "type" bibinfo.check
1023     *
1024   }
1025   if$
1026 }
1027 if$
1028 }

```

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

```

1029 FUNCTION {format.chapter}
1030 { chapter duplicate$ empty$
1031   { pop$ "" }
1032   { "chapter" bibinfo.check
1033     "c." swap$ *
1034   }
1035   if$
1036 }

```

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

```

1037 FUNCTION {format.series.number.chapter}
1038 { series duplicate$ empty$
1039   { pop$ "(" }
1040   { "series" bibinfo.check
1041     "(" swap$ *
1042   }
1043   if$
1044   type duplicate$ empty$
1045   { pop$
1046     number duplicate$ empty$
1047     'pop$
1048     { "number" bibinfo.check
1049       swap$ duplicate$ "(" =
1050       { swap$ * }
1051       { ", " * swap$ * }
1052     }
1053     if$
1054   }
1055   chapter duplicate$ empty$
1056   'pop$
1057   { "chapter" bibinfo.check
1058     swap$ duplicate$ "(" =
1059     { swap$ * }
1060     { ", c." * swap$ * }
1061   }
1062   if$
1063 }
1064 { "type" bibinfo.check
1065   swap$ duplicate$ "(" =
1066   { swap$ * }
1067   { ", " * swap$ * }
1068   if$
1069   number duplicate$ empty$
1070   { pop$
1071     chapter duplicate$ empty$
1072     'pop$
1073     { "chapter" bibinfo.check
1074       " c." swap$ * *
1075     }
1076     if$
1077   }
1078   { "number" bibinfo.check
1079     " " swap$ * *
1080     chapter duplicate$ empty$
1081     'pop$
1082     { "chapter" bibinfo.check
1083       ", c." swap$ * *
1084     }
1085     if$
1086   }
1087   if$
1088 }
1089 }
1090 if$
1091 ")" *
1092 duplicate$ "(" =
1093 { pop$ "" }
1094 'skip$
1095 if$
1096 }

```

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'

```

1097 FUNCTION {format.case.number}
1098 { casenumber duplicate$ empty$
1099   { pop$
1100     number duplicate$ empty$
1101     { pop$ "" }
1102     { institution field.or.null "Commission" =
1103       userb empty$ not
1104       and
1105       { userb "userb" bibinfo.check

```

```

1106         (" swap$ * ") " * bbl.commission * " " *
1107         swap$ "number" bibinfo.check *
1108     }
1109     { "number" bibinfo.check
1110     (" swap$ * ") *
1111     }
1112     if$
1113 }
1114 if$
1115 }
1116 { "casenumber" bibinfo.check
1117 (" swap$ * ") *
1118 institution field.or.null "Commission" =
1119 number empty$ not
1120 and
1121 { " " * bbl.commission * " " *
1122   number "number" bibinfo.check *
1123 }
1124 'skip$
1125 if$
1126 }
1127 if$
1128 }

```

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

```

1129 FUNCTION {eu.case.check}
1130 { journal field.or.null "OJ" =
1131   number empty$ not
1132   or
1133   casenumber empty$ not
1134   or
1135   'continue.clause
1136   'new.block
1137   if$
1138 }

```

5.9 Notes

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

```

1139 FUNCTION {format.note}
1140 { note empty$
1141   { "" }
1142   { note #1 #1 substring$
1143     duplicate$ "{" =
1144     'skip$
1145     { output.state mid.sentence =
1146       output.state mid.clause =
1147       or
1148         { "l" }
1149         { "u" }
1150     if$
1151     change.case$
1152   }
1153   if$
1154   note #2 global.max$ substring$ * "note" bibinfo.check
1155 }
1156 if$
1157 }

```

5.10 Addresses and publishers

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

```

1158 FUNCTION {format.org.or.pub}
1159 { 't :=
1160   ""
1161   address empty$ t empty$ and
1162   'skip$
1163   { address "address" bibinfo.check *
1164     t empty$
1165     'skip$
1166     { address empty$
1167       'skip$
1168       { ": " * }
1169       if$
1170       t *
1171     }
1172     if$
1173   }
1174   if$
1175 }

```

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

```

1176 FUNCTION {format.publisher.address}
1177 { publisher "publisher" bibinfo.warn format.org.or.pub
1178 }
1179 FUNCTION {format.organization.address}
1180 { publisher empty$
1181   { organization "organization" bibinfo.check format.org.or.pub }
1182   { publisher "publisher" bibinfo.warn format.org.or.pub
1183     organization duplicate$ empty$
1184     'pop$
1185     { "organization" bibinfo.check ". " * swap$ * }
1186     if$
1187   }
1188   if$
1189 }
1190 FUNCTION {format.institution.address}
1191 { publisher empty$
1192   { institution "institution" bibinfo.check format.org.or.pub }
1193   { publisher "publisher" bibinfo.warn format.org.or.pub
1194     institution duplicate$ empty$
1195     'pop$
1196     { "institution" bibinfo.check ". " * swap$ * }
1197     if$
1198   }
1199   if$
1200 }

```

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

```

1201 FUNCTION {format.library}
1202 { library "library" bibinfo.check
1203   emphasize
1204   %<bst>
1205   duplicate$ empty$ not
1206   is.online
1207   and
1208   journal empty$
1209   and
1210   { " " * bbl.online * }
1211   'skip$
1212   if$
1213   %</bst>
1214 }

```

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

```

1215 FUNCTION {format.img.library}
1216 { library duplicate$ empty$

```

```

1217     { pop$ "" }
1218     { "library" bibinfo.check
1219       "At: " swap$ *
1220     }
1221   if$
1222 }
1223 FUNCTION {format.img.library.address}
1224 { address empty$ library empty$ and institution empty$ and
1225   { "" }
1226   { "At: "
1227     address "address" bibinfo.check *
1228     library duplicate$ empty$
1229     { institution duplicate$ empty$
1230       { pop$ "" }
1231       { address empty$
1232         'skip$
1233         { ". " swap$ }
1234         if$
1235         "institution" bibinfo.check * *
1236       }
1237     }
1238   }
1239   { address empty$
1240     'skip$
1241     { ". " swap$ }
1242     if$
1243     "library" bibinfo.check * *
1244   }
1245   if$
1246 }
1247 if$
1248 }
1249 FUNCTION {format.pub.org.lib.address}
1250 { publisher duplicate$ empty$
1251   { pop$
1252     organization duplicate$ empty$
1253     { pop$
1254       format.img.library.address }
1255     { "organization" bibinfo.check
1256       library empty$
1257       { format.org.or.pub }
1258       { ". " *
1259         format.img.library.address *
1260       }
1261     }
1262   }
1263   if$
1264 }
1265 { "publisher" bibinfo.warn format.org.or.pub
1266   library empty$
1267   'skip$
1268   { ". " *
1269     format.img.library *
1270   }
1271   if$
1272 }
1273 if$
1274 }

```

5.11 Pinpointing

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

```

1276 INTEGERS { multiresult }
1277 FUNCTION {multi.page.check}
1278 { 't :=
1279   #0 'multiresult :=
1280   { multiresult not
1281     t empty$ not
1282     and

```

```

1283     }
1284     { t #1 #1 substring$
1285       duplicate$ "-" =
1286       swap$ duplicate$ ", " =
1287       swap$ "+" =
1288     or or
1289       { #1 'multiresult := }
1290       { t #2 global.max$ substring$ 't := }
1291     if$
1292   }
1293 while$
1294 multiresult
1295 }

```

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

```

1296 FUNCTION {n.dashify}
1297 { 't :=
1298   ""
1299   { t empty$ not }
1300   { t #1 #1 substring$ "-" =
1301     { t #1 #2 substring$ "--" = not
1302       { "--" *
1303         t #2 global.max$ substring$ 't :=
1304       }
1305       { { t #1 #1 substring$ "-" = }
1306         { "-" *
1307           t #2 global.max$ substring$ 't :=
1308         }
1309       while$
1310     }
1311     if$
1312   }
1313   { t #1 #1 substring$ *
1314     t #2 global.max$ substring$ 't :=
1315   }
1316   if$
1317 }
1318 while$
1319 }

```

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

```

1320 FUNCTION {format.pages}
1321 { pages duplicate$ empty$ 'skip$
1322   { duplicate$ multi.page.check
1323     { bbl.pages swap$
1324       n.dashify
1325     }
1326     { bbl.page swap$
1327     }
1328     if$
1329     "pages" bibinfo.check
1330     *
1331   }
1332   if$
1333 }

```

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.

```

1334 FUNCTION {format.journal.pages}
1335 { pages duplicate$ empty$ 'pop$
1336   { swap$ duplicate$ empty$
1337     { pop$ pop$ format.pages }
1338     { ", " *
1339       swap$
1340       n.dashify
1341       pages multi.page.check
1342       'bbl.pages
1343       'bbl.page
1344     if$
1345     swap$

```



```

1346     "pages" bibinfo.check
1347     * *
1348   }
1349   if$
1350 }
1351 if$
1352 }

```

`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).

```

1353 FUNCTION {format.journal.eid}
1354 { eid "eid" bibinfo.check
1355   duplicate$ empty$ 'pop$
1356   { swap$ duplicate$ empty$ 'skip$
1357     { ", " *
1358     }
1359   if$
1360   swap$ *
1361 }
1362 if$
1363 }

```

`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.

```

1364 FUNCTION {format.vol.num.pages}
1365 { pubstate field.or.null
1366   duplicate$ "inpress" =
1367   { pop$ bbl.inpress }
1368   { duplicate$ "submitted" =
1369     { pop$ bbl.submitted }
1370     { duplicate$ "inpreparation" =
1371       { pop$ bbl.inpreparation }
1372       { pop$ "" }
1373     if$
1374   }
1375   if$
1376 }
1377 if$
1378 duplicate$ empty$
1379 { pop$
1380   volume field.or.null
1381   duplicate$ empty$ 'skip$
1382   { "volume" bibinfo.check
1383   }
1384   if$
1385   number "number" bibinfo.check duplicate$ empty$ 'skip$
1386   { swap$ duplicate$ empty$
1387     { "there's a number but no volume in " cite$ * warning$ }
1388     'skip$
1389     if$
1390     swap$
1391     "(" swap$ * ")" *
1392   }
1393   if$ *
1394   eid empty$
1395   { format.journal.pages }
1396   { format.journal.eid }
1397   if$
1398 }
1399 'skip$
1400 if$
1401 }

```

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

```

1402 FUNCTION {format.chapter.pages}
1403 { chapter empty$
1404   'format.pages
1405   { type empty$
1406     { bbl.chapter }
1407     { type "1" change.case$
1408       "type" bibinfo.check
1409     }
1410     if$
1411     chapter tie.or.space.prefix
1412     "chapter" bibinfo.check
1413     * *
1414     pages empty$
1415     'skip$
1416     { ", " * format.pages * }
1417     if$
1418   }
1419   if$
1420 }

```

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.

```

1421 FUNCTION {format.journal.series.vol.pages}
1422 { journal field.or.null
1423   duplicate$ "QJ" =
1424   { "journal" bibinfo.check
1425     emphasize
1426     series empty$
1427     volume empty$
1428     and
1429     pages empty$
1430     and
1431     'skip$
1432     { " " * }
1433     if$
1434     series "series" bibinfo.check
1435     volume "volume" bibinfo.check *
1436     duplicate$ empty$
1437     'skip$
1438     { pages field.or.null duplicate$ empty$
1439       'skip$
1440       { "/" swap$ *
1441       }
1442       if$ *
1443     }
1444     if$ *
1445   }
1446   { duplicate$ "ECR" =
1447     { "journal" bibinfo.check
1448       volume "volume" bibinfo.check
1449       duplicate$ empty$
1450       'skip$
1451       { " " swap$ * "--" * * }
1452       if$
1453     }
1454     { "journal" bibinfo.check
1455     %< *bst2>
1456     is.online
1457     { " " * bbl.online * }
1458     'skip$
1459     if$
1460     %< /bst2>
1461     volume "volume" bibinfo.check
1462     duplicate$ empty$
1463     'pop$
1464     { " " * swap$ * }
1465     if$
1466     pages empty$
1467     eid empty$
1468     and

```

```

1469         'skip$
1470         { " " * }
1471         if$
1472     }
1473     if$
1474 }
1475 if$
1476 eid empty$
1477 { pages "pages" bibinfo.check * }
1478 { eid "eid" bibinfo.check * }
1479 if$
1480 }

```

5.12 URLs

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

```

1481 FUNCTION {format.eprint}
1482 { eprint duplicate$ empty$
1483   'skip$
1484   { "\eprint"
1485     archive empty$
1486     'skip$
1487     { "[" * archive * "]" * }
1488     if$
1489     "{" * swap$ * "}" *
1490   }
1491   if$
1492 }

```

`format.url` constructs a string Available from: `\url{url} [Accessed urldate]`.

```

1493 FUNCTION {format.url}
1494 { doi empty$
1495   { url }
1496   { "https://doi.org/" doi * }
1497   if$
1498   duplicate$ empty$
1499   { pop$ "" }
1500   { "\urlprefix\url{" swap$ * "}" * }
1501   if$
1502   urlyear empty$
1503   { urldate empty$
1504     'skip$
1505     { "[\urldateprefix{" * urldate * "]" * }
1506     if$
1507   }
1508   { " [\urldateprefix{" * urlyear * "]" * }
1509   if$
1510 }

```

5.13 Cross referencing between entries

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

```

1511 FUNCTION {format.article.crossref}
1512 { word.in
1513   " \cite{" * crossref * "}" *
1514 }
1515 FUNCTION {format.incoll.inproc.crossref}
1516 { word.in
1517   " \cite{" * crossref * "}" *
1518 }

```

`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.

```

1519 FUNCTION {format.book.crossref}
1520 { volume duplicate$ empty$
1521   { "empty volume in " cite$ * "'s crossref of " * crossref * warning$
1522     pop$ word.in
1523   }
1524   { bbl.volume
1525     capitalize
1526     swap$ tie.or.space.prefix "volume" bibinfo.check * * bbl.of space.word *
1527   }
1528   if$
1529   " \cite{" * crossref * "}" *
1530 }

```

5.14 Formatting drivers for specific entry types

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

5.14.1 Article

```

1532 FUNCTION {article}
1533 { output.bibitem
1534   format.authors "author" output.check
1535   author format.key output
1536   format.date "year" output.check
1537   date.block
1538   format.title "title" output.check
1539   get.title.addenda
1540   new.block
1541   crossref missing$
1542     { journal "t" change.case$
1543       "journal" bibinfo.check
1544       emphasize
1545       "journal" output.check
1546     %<*bst2>
1547     is.online
1548       { continue.clause
1549         bbl.online output
1550       }
1551       'skip$
1552     if$
1553     %</bst2>
1554     format.vol.num.pages output
1555   }
1556   { format.article.crossref output.nonnull
1557     format.pages output
1558   }
1559   if$
1560   new.block
1561   format.note output
1562   new.block
1563   format.eprint output
1564   format.url output
1565   fin.entry
1566 }

```

5.14.2 Book

```

1567 FUNCTION {book}
1568 { output.bibitem
1569   author empty$
1570   { editor empty$
1571     { format.btitle "title" output.check
1572       format.date "year" output.check
1573       year empty$ 'skip$ { date.block } if$
1574       get.title.addenda
1575     }
1576     { format.editors "author and editor" output.check
1577       editor format.key output
1578       format.date "year" output.check

```

```

1579     date.block
1580     format.btitle "title" output.check
1581     get.title.addenda
1582   }
1583   if$
1584 }
1585 { format.authors output.nonnull
1586   crossref missing$
1587   { "author and editor" editor either.or.check }
1588   'skip$
1589   if$
1590     format.date "year" output.check
1591     date.block
1592     format.btitle "title" output.check
1593     get.title.addenda
1594   }
1595   if$
1596     format.bvolume output
1597     format.number.series output
1598     new.block
1599     format.edition output
1600     crossref missing$
1601     { new.sentence
1602       format.publisher.address output
1603     }
1604     { new.block
1605       format.book.crossref output.nonnull
1606     }
1607   if$
1608     new.block
1609     format.note output
1610     new.block
1611     format.eprint output
1612     format.url output
1613     fin.entry
1614 }

```

5.14.3 Unpublished or self-published monograph

```

1615 FUNCTION {booklet}
1616 { output.bibitem
1617   author empty$
1618   { format.btitle "title" output.check
1619     format.date "year" output.check
1620     year empty$ 'skip$ { date.block } if$
1621     get.title.addenda
1622   }
1623   { format.authors output
1624     author format.key output
1625     format.date "year" output.check
1626     date.block
1627     format.btitle "title" output.check
1628     get.title.addenda
1629   }
1630   if$
1631     new.block
1632     format.booklet.type output
1633     new.block
1634     howpublished "howpublished" bibinfo.check output
1635     new.block
1636     format.note output
1637     publisher empty$ 'skip$
1638     { new.block}
1639   if$
1640     format.publisher.address output
1641     new.block
1642     format.eprint output
1643     format.url output
1644     fin.entry
1645 }

```

5.14.4 Work in book

```
1646 FUNCTION {inbook}
1647 { output.bibitem
1648   author empty$
1649   { format.editors "author and editor" output.check
1650     editor format.key output
1651   }
1652   { format.authors output.nonnull
1653     crossref missing$
1654     { "author and editor" editor either.or.check }
1655     'skip$
1656     if$
1657   }
1658   if$
1659   format.date "year" output.check
1660   date.block
1661   format.btitle "title" output.check
1662   get.title.addenda
1663   crossref missing$
1664   { format.publisher.address output
1665     format.bvolume output
1666     format.chapter.pages "chapter and pages" output.check
1667     new.block
1668     format.number.series output
1669     new.sentence
1670   }
1671   { format.chapter.pages "chapter and pages" output.check
1672     new.block
1673     format.book.crossref output.nonnull
1674   }
1675   if$
1676   format.edition output
1677   new.block
1678   format.note output
1679   new.block
1680   format.eprint output
1681   format.url output
1682   fin.entry
1683 }
```

5.14.5 Work in collection

```
1684 FUNCTION {incollection}
1685 { output.bibitem
1686   format.authors "author" output.check
1687   author format.key output
1688   format.date "year" output.check
1689   date.block
1690   format.title "title" output.check
1691   get.title.addenda
1692   new.block
1693   crossref missing$
1694   { format.in.ed.booktitle "booktitle" output.check
1695     new.block
1696     format.publisher.address output
1697     format.bvolume output
1698     format.number.series output
1699     format.chapter.pages output
1700     new.sentence
1701     format.edition output
1702   }
1703   { format.incoll.inproc.crossref output.nonnull
1704     format.chapter.pages output
1705   }
1706   if$
1707   new.block
1708   format.note output
1709   new.block
1710   format.eprint output
1711   format.url output
```

```

1712   fin.entry
1713 }

```

5.14.6 Conference paper

```

1714 FUNCTION {inproceedings}
1715 { output.bibitem
1716   format.authors "author" output.check
1717   author format.key output
1718   format.date "year" output.check
1719   date.block
1720   format.title "title" output.check
1721   get.title.addenda
1722   new.block
1723   crossref missing$
1724     { format.in.ed.booktitle "booktitle" output.check
1725       eventyear output
1726       venue output
1727       new.sentence
1728       publisher empty$
1729         { format.organization.address output }
1730         { organization "organization" bibinfo.check output
1731           format.publisher.address output
1732         }
1733       if$
1734       format.bvolume output
1735       format.number.series output
1736       format.pages output
1737     }
1738     { format.incoll.inproc.crossref output.nonnull
1739       format.pages output
1740     }
1741   if$
1742   new.block
1743   format.note output
1744   new.block
1745   format.eprint output
1746   format.url output
1747   fin.entry
1748 }
1749 FUNCTION {conference} { inproceedings }

```

5.14.7 Legal case studies

```

1750 FUNCTION {jurisdiction}
1751 { output.bibitem
1752   author empty$
1753   { format.btitle "title" output.check
1754     continue.clause
1755     format.case.number output
1756     number empty$
1757     casenumber empty$
1758     and
1759     'new.sentence
1760     'continue.clause
1761   if$
1762   format.jur.date "year" output.check
1763   year empty$ 'skip$ { eu.case.check } if$
1764   get.title.addenda
1765 }
1766 { format.authors output
1767   author format.key output
1768   format.jur.date "year" output.check
1769   date.block
1770   continue.clause
1771   format.btitle "title" output.check
1772   get.title.addenda
1773 }
1774 if$

```

```

1775 eu.case.check
1776 format.note output
1777 note empty$
1778 'eu.case.check
1779 'new.block
1780 if$
1781 crossref missing$
1782 { continue.clause
1783   format.journal.series.vol.pages output
1784 }
1785 { format.article.crossref output.nonnull
1786   format.pages output
1787 }
1788 if$
1789 new.block
1790 format.eprint output
1791 format.url output
1792 fin.entry
1793 }

```

5.14.8 Legislation

```

1794 FUNCTION {uklegislation}
1795 { output.bibitem
1796   author empty$
1797   { format.btitle "title" output.check
1798     continue.clause
1799     format.date emphasize "year" output.check
1800     get.title.addenda
1801   }
1802   { format.authors output
1803     author format.key output
1804     format.date "year" output.check
1805     date.block
1806     continue.clause
1807     format.btitle "title" output.check
1808     get.title.addenda
1809   }
1810   if$
1811   entrysubtype field.or.null "secondary" =
1812   { number "number" bibinfo.check output }
1813   { series empty$ type empty$ and
1814     { number empty$
1815       { format.chapter output }
1816       { continue.clause
1817         format.series.number.chapter output
1818       }
1819       if$
1820     }
1821     { chapter empty$
1822       { new.block }
1823       { continue.clause }
1824       if$
1825       format.series.number.chapter output
1826     }
1827     if$
1828   }
1829   if$
1830   note empty$
1831   'skip$
1832   { new.block
1833     format.note output
1834     new.block
1835   }
1836   if$
1837   publisher empty$ 'skip$
1838   { entrysubtype field.or.null "secondary" =
1839     'skip$
1840     { new.block }
1841     if$
1842     format.publisher.address output

```



```

1843     }
1844     if$
1845     new.block
1846     format.eprint output
1847     format.url output
1848     fin.entry
1849 }
1850 FUNCTION {eulegislation}
1851 { output.bibitem
1852   author empty$
1853   { format.title "title" output.check
1854     continue.clause
1855     format.eu.date "year" output.check
1856     get.title.addenda
1857   }
1858   { format.authors output
1859     author format.key output
1860     format.eu.date "year" output.check
1861     date.block
1862     continue.clause
1863     format.title "title" output.check
1864     get.title.addenda
1865   }
1866   if$
1867   eu.case.check
1868   format.note output
1869   note empty$
1870   'eu.case.check
1871   'new.block
1872   if$
1873   crossref missing$
1874   { continue.clause
1875     format.journal.series.vol.pages output
1876   }
1877   { format.article.crossref output.nonnull
1878     format.pages output
1879   }
1880   if$
1881   new.block
1882   format.eprint output
1883   format.url output
1884   fin.entry
1885 }
1886 FUNCTION {legislation}
1887 { journal empty$
1888   'uklegislation
1889   'eulegislation
1890   if$
1891 }

```

5.14.9 Manual

```

1892 FUNCTION {manual}
1893 { output.bibitem
1894   author empty$
1895   { format.btitle "title" output.check
1896     format.date "year" output.check
1897     year empty$ 'skip$ { date.block } if$
1898     get.title.addenda
1899   }
1900   { format.authors output
1901     author format.key output
1902     format.date "year" output.check
1903     date.block
1904     format.btitle "title" output.check
1905     get.title.addenda
1906   }
1907   if$
1908   format.edition output
1909   new.block
1910   format.manual.number output

```

```

1911 new.block
1912 format.library output
1913 organization address new.block.checkb
1914 format.organization.address output
1915 new.block
1916 format.note output
1917 new.block
1918 format.eprint output
1919 format.url output
1920 fin.entry
1921 }

```

5.14.10 Image

```

1922 FUNCTION {image}
1923 { output.bibitem
1924   author empty$
1925   { format.btitle "title" output.check
1926     format.date "year" output.check
1927     year empty$ 'skip$ { date.block } if$
1928     get.title.addenda
1929   }
1930   { format.authors output
1931     author format.key output
1932     format.date "year" output.check
1933     date.block
1934     format.btitle "title" output.check
1935     get.title.addenda
1936   }
1937   if$
1938   format.edition output
1939   new.block
1940   format.manual.number output
1941   new.block
1942   format.pub.org.lib.address output
1943   new.block
1944   format.note output
1945   new.block
1946   format.eprint output
1947   format.url output
1948   fin.entry
1949 }

```

5.14.11 Master's thesis

```

1950 FUNCTION {mastersthesis}
1951 { output.bibitem
1952   format.authors "author" output.check
1953   author format.key output
1954   format.date "year" output.check
1955   date.block
1956   format.btitle "title" output.check
1957   get.title.addenda
1958   new.block
1959   bbl.mthesis format.thesis.type output.nonnull
1960   new.block
1961   school "school" bibinfo.warn output
1962   address "address" bibinfo.check output
1963   new.block
1964   format.note output
1965   new.block
1966   format.eprint output
1967   format.url output
1968   fin.entry
1969 }

```

5.14.12 Miscellaneous

```
1970 FUNCTION {misc}
1971 { output.bibitem
1972   author empty$
1973   { format.title "title" output.check
1974     format.date "year" output.check
1975     year empty$ 'skip$ { date.block } if$
1976     get.title.addenda
1977   }
1978   { format.authors output
1979     author format.key output
1980     format.date "year" output.check
1981     date.block
1982     format.title "title" output.check
1983     get.title.addenda
1984   }
1985   if$
1986   new.block
1987   howpublished "howpublished" bibinfo.check output
1988   new.block
1989   format.note output
1990   new.block
1991   format.eprint output
1992   format.url output
1993   fin.entry
1994 }
```

5.14.13 Doctoral thesis

```
1995 FUNCTION {phdthesis}
1996 { output.bibitem
1997   format.authors "author" output.check
1998   author format.key output
1999   format.date "year" output.check
2000   date.block
2001   format.btitle "title" output.check
2002   get.title.addenda
2003   new.block
2004   bbl.phdthesis format.thesis.type output.nonnull
2005   new.block
2006   school "school" bibinfo.warn output
2007   address "address" bibinfo.check output
2008   new.block
2009   format.note output
2010   new.block
2011   format.eprint output
2012   format.url output
2013   fin.entry
2014 }
```

5.14.14 Conference proceedings

```
2015 FUNCTION {proceedings}
2016 { output.bibitem
2017   editor empty$
2018   { format.btitle "title" output.check
2019     format.date "year" output.check
2020     year empty$ 'skip$ { date.block } if$
2021     get.title.addenda
2022   }
2023   { format.editors output
2024     editor format.key output
2025     format.date "year" output.check
2026     date.block
2027     format.btitle "title" output.check
2028     get.title.addenda
2029   }
2030   if$
2031   format.bvolume output
```

```

2032 format.number.series output
2033 new.sentence
2034 publisher empty$
2035   { format.organization.address output }
2036   { organization "organization" bibinfo.check output
2037     format.publisher.address output
2038   }
2039 if$
2040 new.block
2041 format.note output
2042 new.block
2043 format.eprint output
2044 format.url output
2045 fin.entry
2046 }

```

5.14.15 Report

```

2047 FUNCTION {techreport}
2048 { output.bibitem
2049   author empty$
2050   { format.btitle "title" output.check
2051     format.date "year" output.check
2052     year empty$ 'skip$ { date.block } if$
2053     get.title.addenda
2054   }
2055   { format.authors output
2056     author format.key output
2057     format.date "year" output.check
2058     date.block
2059     format.btitle "title" output.check
2060     get.title.addenda
2061   }
2062 if$
2063 new.block
2064 format.tr.number output
2065 new.block
2066 publisher empty$
2067   { format.institution.address output }
2068   { institution "institution" bibinfo.check output
2069     format.publisher.address output
2070   }
2071 if$
2072 new.block
2073 format.library output
2074 new.block
2075 format.note output
2076 new.block
2077 format.eprint output
2078 format.url output
2079 fin.entry
2080 }

```

5.14.16 Unpublished works

```

2081 FUNCTION {unpublished}
2082 { output.bibitem
2083   author empty$
2084   { booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2085     format.date "year" output.check
2086     year empty$ 'skip$ { date.block } if$
2087     get.title.addenda
2088   }
2089   { format.authors output
2090     author format.key output
2091     format.date "year" output.check
2092     date.block
2093     booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2094     get.title.addenda

```

```

2095     }
2096     if$
2097     new.block
2098     format.in.ed.booktitle output
2099     new.block
2100     howpublished "howpublished" bibinfo.check output
2101     new.block
2102     bbl.unpublished output
2103     new.block
2104     format.note output
2105     new.block
2106     format.eprint output
2107     format.url output
2108     fin.entry
2109 }

```

5.14.17 Aliases and fallbacks

Here are a list of aliases supported by this style.

```

2111 FUNCTION {letter}
2112 { journal empty$
2113   'unpublished
2114   'article
2115   if$
2116 }
2117 FUNCTION {audio} { booklet }
2118 FUNCTION {movie} { booklet }
2119 FUNCTION {music} { booklet }
2120 FUNCTION {video} { booklet }
2121 FUNCTION {dataset} { manual }
2122 FUNCTION {electronic} { manual }
2123 FUNCTION {online} { manual }
2124 FUNCTION {patent} { manual }
2125 FUNCTION {software} { manual }
2126 FUNCTION {standard} { manual }
2127 FUNCTION {www} { manual }
2128 FUNCTION {thesis} { phdthesis }
2129 FUNCTION {report} { techreport }

```

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

```

2130 FUNCTION {default.type} { misc }

```

5.15 Bibliography processing

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

```

2132 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.

```

2133 FUNCTION {sortify}
2134 { purify$
2135   "l" change.case$
2136 }

```

`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`.

```

2137 INTEGERS { len }
2138 FUNCTION {chop.word}
2139 { 's :=
2140   'len :=
2141   s #1 len substring$ =
2142     { s len #1 + global.max$ substring$ }
2143     's
2144   if$
2145 }

```

`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.'

```

2146 FUNCTION {format.lab.names}
2147 { 's :=
2148   "" 't :=
2149   #1 'nameptr :=
2150   s num.names$ 'numnames :=
2151   numnames 'namesleft :=
2152   { namesleft #0 > }
2153   { s nameptr
2154     "{vv~}{ll}" format.name$
2155     't :=
2156     nameptr #1 >
2157     { nameptr #2 =
2158       numnames #3 > and
2159       { "others" 't :=
2160         #1 'namesleft := }
2161       'skip$
2162       if$
2163       namesleft #1 >
2164       { ", " * t * }
2165       { s nameptr "{ll}" format.name$ duplicate$ "others" =
2166         { 't := }
2167         { pop$ }
2168         if$
2169         t "others" =
2170         { " " * bbl.etal *
2171         }
2172         { bbl.and
2173           space.word * t *
2174         }
2175         if$
2176       }
2177       if$
2178     }
2179     't
2180     if$
2181     nameptr #1 + 'nameptr :=
2182     namesleft #1 - 'namesleft :=
2183   }
2184   while$
2185 }

```

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

```

2186 FUNCTION {author.editor.key.label}
2187 { author empty$
2188   { editor empty$
2189     { key empty$
2190       { title empty$
2191         { cite$ #1 #3 substring$ }
2192         { title
2193           type$ "book" =
2194             { emphasize }
2195             'skip$
2196           if$
2197         }
2198         if$
2199       }
2200       'key
2201       if$
2202     }
2203     { editor format.lab.names }
2204     if$
2205   }
2206   { author format.lab.names }
2207   if$
2208 }

```

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

```

2209 FUNCTION {editor.key.label}
2210 { editor empty$
2211   { key empty$
2212     { title empty$
2213       { cite$ #1 #3 substring$ }
2214       { title emphasize }
2215       if$
2216     }
2217     'key
2218     if$
2219   }
2220   { editor format.lab.names }
2221   if$
2222 }

```

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

```

2223 FUNCTION {author.key.label}
2224 { author empty$
2225   { key empty$
2226     { title empty$
2227       { cite$ #1 #3 substring$ }
2228       { title
2229         type$ "article" =
2230         type$ "incollection" =
2231         or
2232         type$ "inproceedings" =
2233         or
2234         type$ "misc" =
2235         or
2236         'skip$
2237         { emphasize }
2238         if$
2239       }
2240     }
2241   }
2242   'key
2243   if$
2244 }
2245 { author format.lab.names }
2246 if$
2247 }

```

calc.short.authors chooses between the above.

```

2248 FUNCTION {calc.short.authors}
2249 { type$ "book" =
2250   type$ "inbook" =
2251   or
2252   'author.editor.key.label
2253   { type$ "proceedings" =
2254     'editor.key.label
2255     'author.key.label
2256     if$
2257   }
2258   if$
2259   'short.list :=
2260 }

```

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

```

2261 FUNCTION {calc.label}
2262 { calc.short.authors
2263   short.list
2264   "("
2265   *
2266   year nodate.check duplicate$ empty$
2267   { pop$ "" }
2268   'skip$
2269   if$
2270   *

```

```
2271 'label :=
2272 }
```

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.’

```
2273 FUNCTION {sort.format.names}
2274 { 's :=
2275 #1 'nameptr :=
2276 ""
2277 s num.names$ 'numnames :=
2278 numnames 'namesleft :=
2279 { namesleft #0 > }
2280 { s nameptr
2281 "{ll{ }}{ f{ }}{ jj{ }}{ vv{ }}"
2282 format.name$ 't :=
2283 nameptr #1 >
2284 { " " *
2285 namesleft #1 =
2286 t "others" =
2287 and
2288 { "zzzzz" 't := }
2289 'skip$
2290 if$
2291 t sortify *
2292 }
2293 { t sortify * }
2294 if$
2295 nameptr #1 + 'nameptr :=
2296 namesleft #1 - 'namesleft :=
2297 }
2298 while$
2299 }
```

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

```
2300 FUNCTION {sort.format.title}
2301 { 't :=
2302 "A " #2
2303 "An " #3
2304 "The " #4 t chop.word
2305 chop.word
2306 chop.word
2307 sortify
2308 #1 global.max$ substring$
2309 }
```

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.

```
2310 FUNCTION {author.editor.sort}
2311 { author empty$
2312 { editor empty$
2313 { title empty$
2314 { key empty$
2315 { "to sort, need author, editor, title, or key in " cite$ * warning$
2316 ""
2317 }
2318 { key sortify }
2319 if$
2320 }
2321 { title sort.format.title }
2322 if$
2323 }
2324 { editor sort.format.names }
2325 if$
2326 }
2327 { author sort.format.names }
2328 if$
2329 }
```


editor.sort does similar but skips the author names.

```
2330 FUNCTION {editor.sort}
2331 { editor empty$
2332   { title empty$
2333     { key empty$
2334       { "to sort, need editor, title, or key in " cite$ * warning$
2335         ""
2336       }
2337     { key sortify }
2338     if$
2339   }
2340   { title sort.format.title }
2341   if$
2342 }
2343 { editor sort.format.names }
2344 if$
2345 }
```

author.sort does similar but skips the editor names.

```
2346 FUNCTION {author.sort}
2347 { author empty$
2348   { title empty$
2349     { key empty$
2350       { "to sort, need author, title, or key in " cite$ * warning$
2351         ""
2352       }
2353     { key sortify }
2354     if$
2355   }
2356   { title sort.format.title }
2357   if$
2358 }
2359 { author sort.format.names }
2360 if$
2361 }
```

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.

```
2363 FUNCTION {presort}
2364 { calc.label
2365   label sortify
2366   " "
2367   *
2368   type$ "book" =
2369   type$ "inbook" =
2370   or
2371   'author.editor.sort
2372   { type$ "proceedings" =
2373     'editor.sort
2374     'author.sort
2375     if$
2376   }
2377   if$
2378   #1 entry.max$ substring$
2379   'sort.label :=
2380   sort.label
2381   *
2382   " "
2383   *
2384   title field.or.null
2385   sort.format.title
2386   *
2387   #1 entry.max$ substring$
2388   'sort.key$ :=
2389 }
```

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.

```
2391 ITERATE {presort}
2392 SORT
```

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

```
2393 STRINGS { last.label next.extra }
2394 INTEGERS { last.extra.num last.extra.num.extended last.extra.num.blank number.label }
2395 FUNCTION {initialize.extra.label.stuff}
2396 { #0 int.to.chr$ 'last.label :=
2397   "" 'next.extra :=
2398   #0 'last.extra.num :=
2399   "a" chr.to.int$ #1 - 'last.extra.num.blank :=
2400   last.extra.num.blank 'last.extra.num.extended :=
2401   #0 'number.label :=
2402 }
2403 FUNCTION {forward.pass}
2404 { last.label label =
2405   { last.extra.num #1 + 'last.extra.num :=
2406     last.extra.num "z" chr.to.int$ >
2407     { "a" chr.to.int$ 'last.extra.num :=
2408       last.extra.num.extended #1 + 'last.extra.num.extended :=
2409     }
2410     'skip$
2411     if$
2412     last.extra.num.extended last.extra.num.blank >
2413     { last.extra.num.extended int.to.chr$
2414       last.extra.num int.to.chr$
2415       * 'extra.label := }
2416     { last.extra.num int.to.chr$ 'extra.label := }
2417     if$
2418   }
2419   { "a" chr.to.int$ 'last.extra.num :=
2420     "" 'extra.label :=
2421     label 'last.label :=
2422   }
2423   if$
2424   number.label #1 + 'number.label :=
2425 }
2426 FUNCTION {reverse.pass}
2427 { next.extra "b" =
2428   { "a" 'extra.label := }
2429   'skip$
2430   if$
2431   extra.label 'next.extra :=
2432   extra.label
2433   duplicate$ empty$
2434   'skip$
2435   { "{\natexlab{" swap$ * "}" * }
2436   if$
2437   'extra.label :=
2438   label extra.label * 'label :=
2439 }
```

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.

```
2440 EXECUTE {initialize.extra.label.stuff}
2441 ITERATE {forward.pass}
2442 REVERSE {reverse.pass}
2443 FUNCTION {bib.sort.order}
2444 { sort.label
2445   " "
2446   *
2447   year field.or.null sortify
2448   *
2449   " "
2450   *
2451   title field.or.null
2452   sort.format.title
2453   *
2454   #1 entry.max$ substring$
```

```

2455 'sort.key$ :=
2456 }
2457 ITERATE {bib.sort.order}
2458 SORT

```

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

```

2459 FUNCTION {begin.bib}
2460 { preamble$ empty$
2461   'skip$
2462   { preamble$ write$ newline$ }
2463   if$
2464   "\begin{thebibliography}{ number.label int.to.str$ * }" *
2465   write$ newline$
2466   "\providecommand{\natexlab}[1]{#1}"
2467   write$ newline$
2468   "\providecommand{\url}[1]{\texttt{#1}}"
2469   write$ newline$
2470   "\providecommand{\urlprefix}{Available from: }"
2471   write$ newline$
2472   "\providecommand{\urldateprefix}{Accessed }"
2473   write$ newline$
2474   "\providecommand{\selectlanguage}[1]{\relax}"
2475   write$ newline$
2476   "\providecommand{\bibinfo}[2]{#2}"
2477   write$ newline$
2478   "\providecommand{\eprint}[2][ ]{\url{#2}}"
2479   write$ newline$
2480 }
2481 EXECUTE {begin.bib}
2482 EXECUTE {init.state.consts}

```

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

```

2483 ITERATE {call.type$}

```

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

```

2484 FUNCTION {end.bib}
2485 { newline$
2486   "\end{thebibliography}" write$ newline$
2487 }
2488 EXECUTE {end.bib}

```