

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

Maintainer: Alex Ball\*

Package v3.0 – 10 July 2019

## 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<sub>T</sub>E<sub>X</sub>. 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](https://commons.wikimedia.org/wiki/File:Royal_Crescent_in_Bath,_England_-_July_2006.jpg) [Accessed 7 January 2016].

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

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


RI

 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<sub>T</sub>E<sub>X</sub> 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](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](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](mailto:lis-link@jiscmail.ac.uk) [Accessed 30 July 2004].

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

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

RX

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Internally, a letter entry with a journal field is treated like an article entry, while one without is treated like an unpublished entry.

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

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

(Clark, 2004)

### Database

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

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

```
@online{bvd2008bt,  
  author = {{Bureau van Dijk}},  
  year = {2008},  
  title = {{BT} {Group} PLC Company Report},  
  library = {FAME},  
  address = {London},  
  organization = {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].

```
@online{wilson2013rgc,  
  author = {Wilson, D.},  
  year = {2013},  
  title = {Real Geometry and Connectedness via Triangular Description: {CAD} Example Bank},  
  address = {Bath},  
  organization = {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](http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi_20121916_en.pdf) [Accessed 17 April 2016].
-  *The Human Medicines Regulations 2012* [Online], No.1916, United Kingdom: HMSO. Available from: [http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi\\_20121916\\_en.pdf](http://www.legislation.gov.uk/uksi/2012/1916/pdfs/uksi_20121916_en.pdf) [Accessed 17 April 2016].



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

RX

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

## 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>	<p>(Council Regulation [EC] 1984/2003)</p>
--	--

## 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](mailto: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](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](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](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<sub>T</sub>E<sub>X</sub> ones plus `archive`, `doi`, `eid`, `eprint`, `entrysubtype`, `eventyear`, `keywords`, `language`, `library`, `options`, `pubstate`, `sortyear`, `titleaddon`, `translator`, `url`, `urldate`, `urlyear`, `userb`, `venue`, and `version`, all of which have been backported from `natbib` and `biblatex`. `casenumber` is specific to this style.

```
16  ENTRY
17  { address
18  archive
19  author
20  booktitle
21  casenumber
22  chapter
23  doi
24  edition
25  editor
26  eid
27  entrysubtype
28  eprint
29  eventyear
30  howpublished
31  institution
32  journal
33  key
34  keywords
35  language
36  library
37  month
38  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$
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   { bbl.online output }
674   'skip$
675   if$
676   continue.clause
677   %</bst2>
678   format.translators output
679 }

```

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

```

680 FUNCTION {word.in}
681 { bbl.in capitalize
682   ":" *
683   " " * }

```

format.booktitle returns the booktitle field in italics.

```

684 FUNCTION {format.booktitle}
685 { booktitle
686   duplicate$ empty$ 'skip$
687   { "t" change.case$ }
688   if$

```

```

689 "booktitle" bibinfo.check
690 emphasize
691 }

```

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

```

692 FUNCTION {format.in.ed.booktitle}
693 { format.booktitle duplicate$ empty$ 'skip$
694   { editor "editor" format.names.ed duplicate$ empty$ 'pop$
695     { "," *
696       " " *
697       get.bbl.editor add.period$
698       " " *
699       * swap$
700       *
701       word.in swap$ *
702     }
703     if$
704   }
705   if$
706 }

```

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

```

707 FUNCTION {legal.journal.shape}
708 { duplicate$ "OJ" =
709   'emphasize
710   'skip$
711   if$
712 }

```

## 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.

```

713 FUNCTION {nodate.check}
714 {
715   %<bst1> skip$
716   %<*bst2>
717   duplicate$ empty$
718   sortyear empty$
719   and
720   { options field.or.null "nodate" =
721     'skip$
722     { pop$ bbl.nodate }
723     if$
724   }
725   'skip$
726   if$
727   %</bst2>
728 }

```

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

```

729 FUNCTION {format.date}
730 { year "year" bibinfo.check nodate.check
731   extra.label *
732 }

```

`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.

```

733 FUNCTION {format.date.brackets}
734 { format.date

```

```

735  "[" swap$ * "]" *
736  }
737  FUNCTION {format.date.parens}
738  { format.date
739    "(" swap$ * ")" *
740  }
741  FUNCTION {format.jur.date.essential}
742  { options field.or.null "scottish-style" =
743    keywords field.or.null "sc" =
744    or
745    'format.date
746    'format.date.brackets
747    if$
748  }
749  FUNCTION {format.jur.date}
750  { journal field.or.null 's :=
751    s "OJ" =
752    s "ECR" =
753    or
754    'format.date.brackets
755    { options field.or.null "year-essential=false" =
756      'format.date.parens
757      { volume empty$
758        'format.jur.date.essential
759        { options field.or.null "year-essential=true" =
760          'format.jur.date.essential
761          'format.date.parens
762          if$
763        }
764      }
765    }
766    if$
767  }
768  if$
769  }

```

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

```

770  FUNCTION {format.eu.date}
771  { journal field.or.null 's :=
772    s "OJ" =
773    s "ECR" =
774    or
775    'format.date.brackets
776    'format.date
777    if$
778  }

```

## 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.

```

779  FUNCTION {format.bvolume}
780  { volume empty$
781    { "" }
782    { bbl.volume volume tie.or.space.prefix
783      "volume" bibinfo.check * *
784      series "series" bibinfo.check
785      duplicate$ empty$ 'pop$
786      { emphasize ", " * swap$ * }
787      if$
788      "volume and number" number either.or.check
789    }
790    if$
791  }

```

`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.

```

792 FUNCTION {format.number.series}
793 { volume empty$
794   { number empty$
795     { series field.or.null }
796     { series empty$
797       { number "number" bibinfo.check }
798       { series "series" bibinfo.check
799         number tie.or.space.prefix "number" bibinfo.check * *
800       }
801     }
802   }
803 }
804 { "" }
805 }
806 if$
807 }

```

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

```

808 FUNCTION {is.num}
809 { chr.to.int$
810   duplicate$ "0" chr.to.int$ < not
811   swap$ "9" chr.to.int$ > not and
812 }

```

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

```

813 FUNCTION {extract.num}
814 { duplicate$ 't :=
815   "" 's :=
816   { t empty$ not }
817   { t #1 #1 substring$
818     t #2 global.max$ substring$ 't :=
819     duplicate$ is.num
820     { s swap$ * 's := }
821     { pop$ "" 't := }
822   }
823 }
824 while$
825 s empty$
826 'skip$
827 { pop$ s }
828 if$
829 }

```

`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.

```

830 FUNCTION {convert.edition}
831 { extract.num "1" change.case$ 's :=
832   s "first" = s "1" = or
833   { bbl.first 't := }
834   { s "second" = s "2" = or
835     { bbl.second 't := }
836     { s "third" = s "3" = or
837       { bbl.third 't := }
838       { s "fourth" = s "4" = or
839         { bbl.fourth 't := }
840         { s "fifth" = s "5" = or
841           { bbl.fifth 't := }
842           { s #1 #1 substring$ is.num
843             { s eng.ord 't := }
844             { edition 't := }
845           }
846         }
847       }
848     }
849   }
850 }
851 if$
852 }

```



```

853     if$
854   }
855   if$
856   t
857 }

```

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

```

858 FUNCTION {format.edition}
859 { edition duplicate$ empty$ 'skip$
860   { convert.edition
861     output.state mid.sentence =
862     { "l" }
863     { "t" }
864     if$ change.case$
865     "edition" bibinfo.check
866     " " * bbl.edition *
867   }
868   if$
869 }

```

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.

```

870 FUNCTION {format.thesis.type}
871 { type duplicate$ empty$
872   'pop$
873   { swap$ pop$
874     "t" change.case$ "type" bibinfo.check
875   }
876   if$
877 }

```

If the type is provided, `format.booklet.type` simply writes the type field with an initial capital. Otherwise it returns an empty string.

```

878 FUNCTION {format.booklet.type}
879 { type duplicate$ empty$
880   { pop$ "" }
881   { "type" bibinfo.check}
882   if$
883 }

```

`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.

```

884 FUNCTION {add.cmd.number}
885 { ". " swap$ "number" bibinfo.check *
886 }
887 FUNCTION {format.legr.number}
888 { series field.or.null 's :=
889   s "C" =
890     'add.cmd.number
891     { s "Cd" =
892       'add.cmd.number
893       { s "Cmd" =
894         'add.cmd.number
895         { s "Cmnd" =
896           'add.cmd.number
897           { s "Cm" =
898             'add.cmd.number
899             { s #1 #3 substring$ 't :=
900               t "HL" =
901                 { ", (" swap$
902                   "number" bibinfo.check *
903                   ")" *
904               }

```

```

905         { ", " swap$
906           "number" bibinfo.check *
907         }
908       if$
909     }
910   if$
911 }
912   if$
913 }
914   if$
915 }
916   if$
917 }
918   if$
919 }
920 FUNCTION {format.tr.number}
921 { series duplicate$ empty$
922   { pop$
923     type duplicate$ empty$
924     { pop$
925       number duplicate$ empty$
926       { pop$ ""
927       }
928       { "(" swap$
929         "number" bibinfo.check
930         * ")" *
931       }
932     if$
933   }
934   { "(" swap$
935     "t" change.case$ "type" bibinfo.check
936     *
937     number duplicate$ empty$
938     'pop$
939     { tie.or.space.prefix
940       "number" bibinfo.check
941       * *
942     }
943     if$
944     ")" *
945   }
946   if$
947 }
948 { "(" swap$ *
949   type duplicate$ empty$
950   { pop$
951     number duplicate$ empty$
952     'pop$
953     { format.legr.number * }
954     if$
955   }
956   { "t" change.case$ "type" bibinfo.check
957     *
958     number duplicate$ empty$
959     'pop$
960     { tie.or.space.prefix
961       "number" bibinfo.check
962       * *
963     }
964     if$
965   }
966   if$
967   ")" *
968   continue.clause
969 }
970   if$
971 }

```

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

```

972 FUNCTION {format.manual.number}
973 { series duplicate$ empty$
974   { pop$
975     type duplicate$ empty$
976     { pop$
977       number duplicate$ empty$
978       { pop$ ""
979       }
980       { "number" bibinfo.check
981       }
982       if$
983     }
984     { "t" change.case$ "type" bibinfo.check
985       number duplicate$ empty$
986       'pop$
987       { tie.or.space.prefix
988         "number" bibinfo.check
989         * *
990       }
991       if$
992     }
993     if$
994   }
995   { number duplicate$ empty$
996     'pop$
997     { tie.or.space.prefix
998       "number" bibinfo.check
999       * *
1000     }
1001     if$
1002     type duplicate$ empty$
1003     'pop$
1004     { ", " swap$ *
1005       "t" change.case$ "type" bibinfo.check
1006       *
1007     }
1008     if$
1009   }
1010   if$
1011 }

```

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

```

1012 FUNCTION {format.chapter}
1013 { chapter duplicate$ empty$
1014   { pop$ "" }
1015   { "chapter" bibinfo.check
1016     "c." swap$ *
1017   }
1018   if$
1019 }

```

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.

```

1020 FUNCTION {format.series.number.chapter}
1021 { series duplicate$ empty$
1022   { pop$ "(" }
1023   { "series" bibinfo.check
1024     "(" swap$ *
1025   }
1026   if$
1027   type duplicate$ empty$
1028   { pop$
1029     number duplicate$ empty$
1030     'pop$
1031     { "number" bibinfo.check
1032       swap$ duplicate$ "(" =
1033       { swap$ * }
1034       { ", " * swap$ * }
1035     }
1036     if$
1037   }

```

```

1037     if$
1038     chapter duplicate$ empty$
1039     'pop$
1040     { "chapter" bibinfo.check
1041       swap$ duplicate$ "(" =
1042         { swap$ * }
1043         { ", c." * swap$ * }
1044     }
1045     }
1046     if$
1047   }
1048   { "type" bibinfo.check
1049     swap$ duplicate$ "(" =
1050     { swap$ * }
1051     { ", " * swap$ * }
1052     if$
1053     number duplicate$ empty$
1054     { pop$
1055       chapter duplicate$ empty$
1056       'pop$
1057       { "chapter" bibinfo.check
1058         " c." swap$ * *
1059       }
1060       if$
1061     }
1062     { "number" bibinfo.check
1063       " " swap$ * *
1064       chapter duplicate$ empty$
1065       'pop$
1066       { "chapter" bibinfo.check
1067         ", c." swap$ * *
1068       }
1069       if$
1070     }
1071     if$
1072   }
1073   if$
1074   ")" *
1075   duplicate$ "(" =
1076   { pop$ "" }
1077   'skip$
1078   if$
1079 }

```

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'

```

1080 FUNCTION {format.case.number}
1081 { casenumber duplicate$ empty$
1082   { pop$
1083     number duplicate$ empty$
1084     { pop$ "" }
1085     { institution field.or.null "Commission" =
1086       userb empty$ not
1087       and
1088       { userb "userb" bibinfo.check
1089         "(" swap$ * " ) " * bbl.commission * " " *
1090         swap$ "number" bibinfo.check *
1091       }
1092       { "number" bibinfo.check
1093         "(" swap$ * " )" *
1094       }
1095     }
1096   }
1097   if$
1098 }
1099 { "casenumber" bibinfo.check
1100   "(" swap$ * " )" *
1101   institution field.or.null "Commission" =
1102   number empty$ not
1103   and
1104   { " " * bbl.commission * " " *
1105     number "number" bibinfo.check *

```

```

1106     }
1107     'skip$
1108     if$
1109     }
1110     if$
1111 }

```

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

```

1112 FUNCTION {eu.case.check}
1113 { journal field.or.null "OJ" =
1114   number empty$ not
1115   or
1116   casenumber empty$ not
1117   or
1118   'continue.clause
1119   'new.block
1120   if$
1121 }

```

## 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).

```

1122 FUNCTION {format.note}
1123 { note empty$
1124   { "" }
1125   { note #1 #1 substring$
1126     duplicate$ "{" =
1127     'skip$
1128     { output.state mid.sentence =
1129       output.state mid.clause =
1130       or
1131       { "l" }
1132       { "u" }
1133       if$
1134       change.case$
1135     }
1136     if$
1137     note #2 global.max$ substring$ * "note" bibinfo.check
1138   }
1139   if$
1140 }

```

## 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.

```

1141 FUNCTION {format.org.or.pub}
1142 { 't :=
1143   ""
1144   address empty$ t empty$ and
1145   'skip$
1146   { address "address" bibinfo.check *
1147     t empty$
1148     'skip$
1149     { address empty$
1150       'skip$
1151       { ": " * }
1152       if$
1153       t *
1154     }
1155     if$
1156   }
1157   if$

```

```
1158 }
```

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

```
1159 FUNCTION {format.publisher.address}
1160 { publisher "publisher" bibinfo.warn format.org.or.pub
1161 }
1162 FUNCTION {format.organization.address}
1163 { publisher empty$
1164   { organization "organization" bibinfo.check format.org.or.pub }
1165   { publisher "publisher" bibinfo.warn format.org.or.pub
1166     organization duplicate$ empty$
1167     'pop$
1168     { "organization" bibinfo.check ". " * swap$ * }
1169     if$
1170   }
1171   if$
1172 }
1173 FUNCTION {format.institution.address}
1174 { publisher empty$
1175   { institution "institution" bibinfo.check format.org.or.pub }
1176   { publisher "publisher" bibinfo.warn format.org.or.pub
1177     institution duplicate$ empty$
1178     'pop$
1179     { "institution" bibinfo.check ". " * swap$ * }
1180     if$
1181   }
1182   if$
1183 }
```

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

```
1184 FUNCTION {format.library}
1185 { library "library" bibinfo.check
1186   emphasize
1187   %< *bst>
1188   duplicate$ empty$ not
1189   is.online
1190   and
1191   journal empty$
1192   and
1193   { " " * bbl.online * }
1194   'skip$
1195   if$
1196   %< /bst>
1197 }
```

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.

```
1198 FUNCTION {format.img.library}
1199 { library duplicate$ empty$
1200   { pop$ "" }
1201   { "library" bibinfo.check
1202     "At: " swap$ *
1203   }
1204   if$
1205 }
1206 FUNCTION {format.img.library.address}
1207 { address empty$ library empty$ and institution empty$ and
1208   { "" }
1209   { "At: "
1210     address "address" bibinfo.check *
1211     library duplicate$ empty$
1212     { institution duplicate$ empty$
1213       { pop$ "" }
1214       { address empty$
1215         'skip$
1216         { ". " swap$ }

```

```

1217         if$
1218         "institution" bibinfo.check * *
1219     }
1220     if$
1221 }
1222 { address empty$
1223     'skip$
1224     { ". " swap$ }
1225     if$
1226     "library" bibinfo.check * *
1227 }
1228 if$
1229 }
1230 if$
1231 }
1232 FUNCTION {format.pub.org.lib.address}
1233 { publisher duplicate$ empty$
1234   { pop$
1235     organization duplicate$ empty$
1236     { pop$
1237       format.img.library.address }
1238     { "organization" bibinfo.check
1239       library empty$
1240       { format.org.or.pub }
1241       { ". " *
1242         format.img.library.address *
1243       }
1244     }
1245   }
1246   if$
1247 }
1248 { "publisher" bibinfo.warn format.org.or.pub
1249   library empty$
1250   'skip$
1251   { ". " *
1252     format.img.library *
1253   }
1254   if$
1255 }
1256 if$
1257 }

```

## 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.

```

1259 INTEGERS { multiresult }
1260 FUNCTION {multi.page.check}
1261 { 't :=
1262   #0 'multiresult :=
1263   { multiresult not
1264     t empty$ not
1265     and
1266   }
1267   { t #1 #1 substring$
1268     duplicate$ "-" =
1269     swap$ duplicate$ "," =
1270     swap$ "+" =
1271     or or
1272     { #1 'multiresult := }
1273     { t #2 global.max$ substring$ 't := }
1274   }
1275 }
1276 while$
1277 multiresult
1278 }

```

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

```

1279 FUNCTION {n.dashify}
1280 { 't :=
1281   ""
1282   { t empty$ not }
1283   { t #1 #1 substring$ "-" =
1284     { t #1 #2 substring$ "--" = not
1285       { "--" *
1286         t #2 global.max$ substring$ 't :=
1287       }
1288       { { t #1 #1 substring$ "-" = }
1289         { "-" *
1290           t #2 global.max$ substring$ 't :=
1291         }
1292         while$
1293       }
1294     if$
1295   }
1296   { t #1 #1 substring$ *
1297     t #2 global.max$ substring$ 't :=
1298   }
1299   if$
1300 }
1301 while$
1302 }

```

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

```

1303 FUNCTION {format.pages}
1304 { pages duplicate$ empty$ 'skip$
1305   { duplicate$ multi.page.check
1306     { bbl.pages swap$
1307       n.dashify
1308     }
1309     { bbl.page swap$
1310     }
1311     if$
1312     "pages" bibinfo.check
1313     *
1314   }
1315   if$
1316 }

```

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.

```

1317 FUNCTION {format.journal.pages}
1318 { pages duplicate$ empty$ 'pop$
1319   { swap$ duplicate$ empty$
1320     { pop$ pop$ format.pages }
1321     { ", " *
1322       swap$
1323       n.dashify
1324       pages multi.page.check
1325       'bbl.pages
1326       'bbl.page
1327       if$
1328       swap$
1329       "pages" bibinfo.check
1330       * *
1331     }
1332     if$
1333   }
1334   if$
1335 }

```

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).

```

1336 FUNCTION {format.journal.eid}
1337 { eid "eid" bibinfo.check

```



```

1338 duplicate$ empty$ 'pop$
1339   { swap$ duplicate$ empty$ 'skip$
1340     { ", " *
1341     }
1342     if$
1343     swap$ *
1344   }
1345   if$
1346 }

```

`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.

```

1347 FUNCTION {format.vol.num.pages}
1348 { pubstate field.or.null
1349   duplicate$ "inpress" =
1350   { pop$ bbl.inpress }
1351   { duplicate$ "submitted" =
1352     { pop$ bbl.submitted }
1353     { duplicate$ "inpreparation" =
1354       { pop$ bbl.inpreparation }
1355       { pop$ "" }
1356     if$
1357   }
1358   if$
1359 }
1360 if$
1361 duplicate$ empty$
1362 { pop$
1363   volume field.or.null
1364   duplicate$ empty$ 'skip$
1365   { "volume" bibinfo.check
1366   }
1367   if$
1368   number "number" bibinfo.check duplicate$ empty$ 'skip$
1369   { swap$ duplicate$ empty$
1370     { "there's a number but no volume in " cite$ * warning$ }
1371     'skip$
1372     if$
1373     swap$
1374     "(" swap$ * ")" *
1375   }
1376   if$ *
1377   eid empty$
1378   { format.journal.pages }
1379   { format.journal.eid }
1380   if$
1381 }
1382 'skip$
1383 if$
1384 }

```

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

```

1385 FUNCTION {format.chapter.pages}
1386 { chapter empty$
1387   'format.pages
1388   { type empty$
1389     { bbl.chapter }
1390     { type "1" change.case$
1391       "type" bibinfo.check
1392     }
1393     if$
1394     chapter tie.or.space.prefix
1395     "chapter" bibinfo.check
1396     * *
1397     pages empty$
1398     'skip$

```

```

1399     { " " * format.pages * }
1400     if$
1401   }
1402   if$
1403 }

```

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.

```

1404 FUNCTION {format.journal.series.vol.pages}
1405 { journal field.or.null
1406   duplicate$ "QJ" =
1407   { "journal" bibinfo.check
1408     emphasize
1409     series empty$
1410     volume empty$
1411     and
1412     pages empty$
1413     and
1414     'skip$
1415     { " " * }
1416     if$
1417     series "series" bibinfo.check
1418     volume "volume" bibinfo.check *
1419     duplicate$ empty$
1420     'skip$
1421     { pages field.or.null duplicate$ empty$
1422       'skip$
1423       { "/" swap$ *
1424         }
1425       if$ *
1426     }
1427     if$ *
1428   }
1429   { duplicate$ "ECR" =
1430     { "journal" bibinfo.check
1431       volume "volume" bibinfo.check
1432       duplicate$ empty$
1433       'skip$
1434       { " " swap$ * "--" * * }
1435       if$
1436     }
1437     { "journal" bibinfo.check
1438     %< *bst>
1439     is.online
1440     { " " * bbl.online * }
1441     'skip$
1442     if$
1443     %< /bst>
1444     volume "volume" bibinfo.check
1445     duplicate$ empty$
1446     'pop$
1447     { " " * swap$ * }
1448     if$
1449     pages empty$
1450     eid empty$
1451     and
1452     'skip$
1453     { " " * }
1454     if$
1455   }
1456   if$
1457 }
1458 if$
1459 eid empty$
1460 { pages "pages" bibinfo.check * }
1461 { eid "eid" bibinfo.check * }
1462 if$
1463 }

```

## 5.12 URLs

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

```
1464 FUNCTION {format.eprint}
1465 { eprint duplicate$ empty$
1466   'skip$
1467   { "\eprint"
1468     archive empty$
1469     'skip$
1470     { "[" * archive * "]" * }
1471     if$
1472     "{" * swap$ * "}" *
1473   }
1474   if$
1475 }
```

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

```
1476 FUNCTION {format.url}
1477 { doi empty$
1478   { url }
1479   { "https://doi.org/" doi * }
1480   if$
1481   duplicate$ empty$
1482   { pop$ "" }
1483   { "\urlprefix\url{" swap$ * "}" * }
1484   if$
1485   urlyear empty$
1486   { urldate empty$
1487     'skip$
1488     { " [\urldateprefix{}]" * urldate * "]" * }
1489     if$
1490   }
1491   { " [\urldateprefix{}]" * urlyear * "]" * }
1492   if$
1493 }
```

## 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).

```
1494 FUNCTION {format.article.crossref}
1495 { word.in
1496   " \cite{" * crossref * "}" *
1497 }
1498 FUNCTION {format.incoll.inproc.crossref}
1499 { word.in
1500   " \cite{" * crossref * "}" *
1501 }
```

`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.

```
1502 FUNCTION {format.book.crossref}
1503 { volume duplicate$ empty$
1504   { "empty volume in " cite$ * "'s crossref of " * crossref * warning$
1505     pop$ word.in
1506   }
1507   { bbl.volume
1508     capitalize
1509     swap$ tie.or.space.prefix "volume" bibinfo.check * * bbl.of space.word *
1510   }
1511   if$
1512   " \cite{" * crossref * "}" *
1513 }
```

## 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

```
1515 FUNCTION {article}
1516 { output.bibitem
1517   format.authors "author" output.check
1518   author format.key output
1519   format.date "year" output.check
1520   date.block
1521   format.title "title" output.check
1522   get.title.addenda
1523   new.block
1524   crossref missing$
1525     { journal "t" change.case$
1526       "journal" bibinfo.check
1527       emphasize
1528       "journal" output.check
1529 %<bst2>
1530     is.online
1531       { continue.clause
1532         bbl.online output
1533       }
1534       'skip$
1535     if$
1536 %</bst2>
1537   format.vol.num.pages output
1538 }
1539 { format.article.crossref output.nonnull
1540   format.pages output
1541 }
1542 if$
1543 new.block
1544 format.note output
1545 new.block
1546 format.eprint output
1547 format.url output
1548 fin.entry
1549 }
```

### 5.14.2 Book

```
1550 FUNCTION {book}
1551 { output.bibitem
1552   author empty$
1553   { editor empty$
1554     { format.btitle "title" output.check
1555       format.date "year" output.check
1556       year empty$ 'skip$ { date.block } if$
1557       get.title.addenda
1558     }
1559     { format.editors "author and editor" output.check
1560       editor format.key output
1561       format.date "year" output.check
1562       date.block
1563       format.btitle "title" output.check
1564       get.title.addenda
1565     }
1566   } if$
1567 }
1568 { format.authors output.nonnull
1569   crossref missing$
1570   { "author and editor" editor either.or.check }
1571   'skip$
1572 } if$
1573 format.date "year" output.check
1574 date.block
1575 format.btitle "title" output.check
1576 get.title.addenda
```

```

1577     }
1578     if$
1579     format.bvolume output
1580     format.number.series output
1581     new.block
1582     format.edition output
1583     crossref missing$
1584     { new.sentence
1585       format.publisher.address output
1586     }
1587     { new.block
1588       format.book.crossref output.nonnull
1589     }
1590     if$
1591     new.block
1592     format.note output
1593     new.block
1594     format.eprint output
1595     format.url output
1596     fin.entry
1597 }

```

### 5.14.3 Unpublished or self-published monograph

```

1598 FUNCTION {booklet}
1599 { output.bibitem
1600   author empty$
1601   { format.btitle "title" output.check
1602     format.date "year" output.check
1603     year empty$ 'skip$ { date.block } if$
1604     get.title.addenda
1605   }
1606   { format.authors output
1607     author format.key output
1608     format.date "year" output.check
1609     date.block
1610     format.btitle "title" output.check
1611     get.title.addenda
1612   }
1613   if$
1614   new.block
1615   format.booklet.type output
1616   new.block
1617   howpublished "howpublished" bibinfo.check output
1618   new.block
1619   format.note output
1620   publisher empty$ 'skip$
1621   { new.block}
1622   if$
1623   format.publisher.address output
1624   new.block
1625   format.eprint output
1626   format.url output
1627   fin.entry
1628 }

```

### 5.14.4 Work in book

```

1629 FUNCTION {inbook}
1630 { output.bibitem
1631   author empty$
1632   { format.editors "author and editor" output.check
1633     editor format.key output
1634   }
1635   { format.authors output.nonnull
1636     crossref missing$
1637     { "author and editor" editor either.or.check }
1638     'skip$
1639   if$
1640 }

```

```

1641  if$
1642  format.date "year" output.check
1643  date.block
1644  format.btitle "title" output.check
1645  get.title.addenda
1646  crossref missing$
1647    { format.publisher.address output
1648      format.bvolume output
1649      format.chapter.pages "chapter and pages" output.check
1650      new.block
1651      format.number.series output
1652      new.sentence
1653    }
1654    { format.chapter.pages "chapter and pages" output.check
1655      new.block
1656      format.book.crossref output.nonnull
1657    }
1658  if$
1659  format.edition output
1660  new.block
1661  format.note output
1662  new.block
1663  format.eprint output
1664  format.url output
1665  fin.entry
1666 }

```

#### 5.14.5 Work in collection

```

1667 FUNCTION {incollection}
1668 { output.bibitem
1669   format.authors "author" output.check
1670   author format.key output
1671   format.date "year" output.check
1672   date.block
1673   format.title "title" output.check
1674   get.title.addenda
1675   new.block
1676   crossref missing$
1677     { format.in.ed.booktitle "booktitle" output.check
1678       new.block
1679       format.publisher.address output
1680       format.bvolume output
1681       format.number.series output
1682       format.chapter.pages output
1683       new.sentence
1684       format.edition output
1685     }
1686     { format.incoll.inproc.crossref output.nonnull
1687       format.chapter.pages output
1688     }
1689   if$
1690   new.block
1691   format.note output
1692   new.block
1693   format.eprint output
1694   format.url output
1695   fin.entry
1696 }

```

#### 5.14.6 Conference paper

```

1697 FUNCTION {inproceedings}
1698 { output.bibitem
1699   format.authors "author" output.check
1700   author format.key output
1701   format.date "year" output.check
1702   date.block
1703   format.title "title" output.check
1704   get.title.addenda

```

```

1705 new.block
1706 crossref missing$
1707   { format.in.ed.booktitle "booktitle" output.check
1708     eventyear output
1709     venue output
1710     new.sentence
1711     publisher empty$
1712     { format.organization.address output }
1713     { organization "organization" bibinfo.check output
1714       format.publisher.address output
1715     }
1716     if$
1717     format.bvolume output
1718     format.number.series output
1719     format.pages output
1720   }
1721   { format.incoll.inproc.crossref output.nonnull
1722     format.pages output
1723   }
1724   if$
1725   new.block
1726   format.note output
1727   new.block
1728   format.eprint output
1729   format.url output
1730   fin.entry
1731 }
1732 FUNCTION {conference} { inproceedings }

```

#### 5.14.7 Legal case studies

```

1733 FUNCTION {jurisdiction}
1734 { output.bibitem
1735   author empty$
1736   { format.btitle "title" output.check
1737     continue.clause
1738     format.case.number output
1739     number empty$
1740     casenumber empty$
1741     and
1742     'new.sentence
1743     'continue.clause
1744     if$
1745     format.jur.date "year" output.check
1746     year empty$ 'skip$ { eu.case.check } if$
1747     get.title.addenda
1748   }
1749   { format.authors output
1750     author format.key output
1751     format.jur.date "year" output.check
1752     date.block
1753     continue.clause
1754     format.btitle "title" output.check
1755     get.title.addenda
1756   }
1757   if$
1758   eu.case.check
1759   format.note output
1760   note empty$
1761   'eu.case.check
1762   'new.block
1763   if$
1764   crossref missing$
1765   { continue.clause
1766     format.journal.series.vol.pages output
1767   }
1768   { format.article.crossref output.nonnull
1769     format.pages output
1770   }
1771   if$
1772   new.block

```

```

1773     format.eprint output
1774     format.url output
1775     fin.entry
1776 }

```

### 5.14.8 Legislation

```

1777 FUNCTION {uklegislation}
1778 { output.bibitem
1779   author empty$
1780   { format.btitle "title" output.check
1781     continue.clause
1782     format.date emphasize "year" output.check
1783     get.title.addenda
1784   }
1785   { format.authors output
1786     author format.key output
1787     format.date "year" output.check
1788     date.block
1789     continue.clause
1790     format.btitle "title" output.check
1791     get.title.addenda
1792   }
1793   if$
1794   entrysubtype field.or.null "secondary" =
1795   { number "number" bibinfo.check output }
1796   { series empty$ type empty$ and
1797     { number empty$
1798       { format.chapter output }
1799       { continue.clause
1800         format.series.number.chapter output
1801       }
1802     }
1803   }
1804   { chapter empty$
1805     { new.block }
1806     { continue.clause }
1807   }
1808   if$
1809   format.series.number.chapter output
1810 }
1811 }
1812 if$
1813 note empty$
1814 'skip$
1815 { new.block
1816   format.note output
1817   new.block
1818 }
1819 if$
1820 publisher empty$ 'skip$
1821 { entrysubtype field.or.null "secondary" =
1822   'skip$
1823   { new.block }
1824   if$
1825   format.publisher.address output
1826 }
1827 if$
1828 new.block
1829 format.eprint output
1830 format.url output
1831 fin.entry
1832 }
1833 FUNCTION {eulegislation}
1834 { output.bibitem
1835   author empty$
1836   { format.title "title" output.check
1837     continue.clause
1838     format.eu.date "year" output.check
1839     get.title.addenda
1840   }

```



```

1841 { format.authors output
1842   author format.key output
1843   format.eu.date "year" output.check
1844   date.block
1845   continue.clause
1846   format.title "title" output.check
1847   get.title.addenda
1848 }
1849 if$
1850 eu.case.check
1851 format.note output
1852 note empty$
1853   'eu.case.check
1854   'new.block
1855 if$
1856 crossref missing$
1857   { continue.clause
1858     format.journal.series.vol.pages output
1859   }
1860   { format.article.crossref output.nonnull
1861     format.pages output
1862   }
1863 if$
1864 new.block
1865 format.eprint output
1866 format.url output
1867 fin.entry
1868 }
1869 FUNCTION {legislation}
1870 { journal empty$
1871   'uklegislation
1872   'eulegislation
1873   if$
1874 }

```

#### 5.14.9 Manual

```

1875 FUNCTION {manual}
1876 { output.bibitem
1877   author empty$
1878   { format.btitle "title" output.check
1879     format.date "year" output.check
1880     year empty$ 'skip$ { date.block } if$
1881     get.title.addenda
1882   }
1883   { format.authors output
1884     author format.key output
1885     format.date "year" output.check
1886     date.block
1887     format.btitle "title" output.check
1888     get.title.addenda
1889   }
1890   if$
1891   format.edition output
1892   new.block
1893   format.manual.number output
1894   new.block
1895   format.library output
1896   organization address new.block.checkb
1897   format.organization.address output
1898   new.block
1899   format.note output
1900   new.block
1901   format.eprint output
1902   format.url output
1903   fin.entry
1904 }

```

#### 5.14.10 Image

```
1905 FUNCTION {image}  
1906 { output.bibitem  
1907   author empty$  
1908   { format.btitle "title" output.check  
1909     format.date "year" output.check  
1910     year empty$ 'skip$ { date.block } if$  
1911     get.title.addenda  
1912   }  
1913   { format.authors output  
1914     author format.key output  
1915     format.date "year" output.check  
1916     date.block  
1917     format.btitle "title" output.check  
1918     get.title.addenda  
1919   }  
1920   if$  
1921   format.edition output  
1922   new.block  
1923   format.manual.number output  
1924   new.block  
1925   format.pub.org.lib.address output  
1926   new.block  
1927   format.note output  
1928   new.block  
1929   format.eprint output  
1930   format.url output  
1931   fin.entry  
1932 }
```

#### 5.14.11 Master's thesis

```
1933 FUNCTION {mastersthesis}  
1934 { output.bibitem  
1935   format.authors "author" output.check  
1936   author format.key output  
1937   format.date "year" output.check  
1938   date.block  
1939   format.btitle "title" output.check  
1940   get.title.addenda  
1941   new.block  
1942   bbl.mthesis format.thesis.type output.nonnull  
1943   new.block  
1944   school "school" bibinfo.warn output  
1945   address "address" bibinfo.check output  
1946   new.block  
1947   format.note output  
1948   new.block  
1949   format.eprint output  
1950   format.url output  
1951   fin.entry  
1952 }
```

#### 5.14.12 Miscellaneous

```
1953 FUNCTION {misc}  
1954 { output.bibitem  
1955   author empty$  
1956   { format.title "title" output.check  
1957     format.date "year" output.check  
1958     year empty$ 'skip$ { date.block } if$  
1959     get.title.addenda  
1960   }  
1961   { format.authors output  
1962     author format.key output  
1963     format.date "year" output.check  
1964     date.block  
1965     format.title "title" output.check  
1966     get.title.addenda
```

```

1967     }
1968     if$
1969     new.block
1970     howpublished "howpublished" bibinfo.check output
1971     new.block
1972     format.note output
1973     new.block
1974     format.eprint output
1975     format.url output
1976     fin.entry
1977 }

```

### 5.14.13 Doctoral thesis

```

1978 FUNCTION {phdthesis}
1979 { output.bibitem
1980   format.authors "author" output.check
1981   author format.key output
1982   format.date "year" output.check
1983   date.block
1984   format.btitle "title" output.check
1985   get.title.addenda
1986   new.block
1987   bbl.phdthesis format.thesis.type output.nonnull
1988   new.block
1989   school "school" bibinfo.warn output
1990   address "address" bibinfo.check output
1991   new.block
1992   format.note output
1993   new.block
1994   format.eprint output
1995   format.url output
1996   fin.entry
1997 }

```

### 5.14.14 Conference proceedings

```

1998 FUNCTION {proceedings}
1999 { output.bibitem
2000   editor empty$
2001     { format.btitle "title" output.check
2002       format.date "year" output.check
2003       year empty$ 'skip$ { date.block } if$
2004       get.title.addenda
2005     }
2006   { format.editors output
2007     editor format.key output
2008     format.date "year" output.check
2009     date.block
2010     format.btitle "title" output.check
2011     get.title.addenda
2012   }
2013   if$
2014   format.bvolume output
2015   format.number.series output
2016   new.sentence
2017   publisher empty$
2018     { format.organization.address output }
2019     { organization "organization" bibinfo.check output
2020       format.publisher.address output
2021     }
2022   if$
2023   new.block
2024   format.note output
2025   new.block
2026   format.eprint output
2027   format.url output
2028   fin.entry
2029 }

```

### 5.14.15 Report

```
2030 FUNCTION {techreport}
2031 { output.bibitem
2032   author empty$
2033   { format.btitle "title" output.check
2034     format.date "year" output.check
2035     year empty$ 'skip$ { date.block } if$
2036     get.title.addenda
2037   }
2038   { format.authors output
2039     author format.key output
2040     format.date "year" output.check
2041     date.block
2042     format.btitle "title" output.check
2043     get.title.addenda
2044   }
2045   if$
2046   new.block
2047   format.tr.number output
2048   new.block
2049   publisher empty$
2050   { format.institution.address output }
2051   { institution "institution" bibinfo.check output
2052     format.publisher.address output
2053   }
2054   if$
2055   new.block
2056   format.library output
2057   new.block
2058   format.note output
2059   new.block
2060   format.eprint output
2061   format.url output
2062   fin.entry
2063 }
```

### 5.14.16 Unpublished works

```
2064 FUNCTION {unpublished}
2065 { output.bibitem
2066   author empty$
2067   { booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2068     format.date "year" output.check
2069     year empty$ 'skip$ { date.block } if$
2070     get.title.addenda
2071   }
2072   { format.authors output
2073     author format.key output
2074     format.date "year" output.check
2075     date.block
2076     booktitle empty$ 'format.btitle 'format.title if$ "title" output.check
2077     get.title.addenda
2078   }
2079   if$
2080   new.block
2081   format.in.ed.booktitle output
2082   new.block
2083   howpublished "howpublished" bibinfo.check output
2084   new.block
2085   bbl.unpublished output
2086   new.block
2087   format.note output
2088   new.block
2089   format.eprint output
2090   format.url output
2091   fin.entry
2092 }
```

### 5.14.17 Aliases and fallbacks

Here are a list of aliases supported by this style.

```
2094 FUNCTION {letter}
2095 { journal empty$
2096   'unpublished
2097   'article
2098   if$
2099 }
2100 FUNCTION {audio} { booklet }
2101 FUNCTION {movie} { booklet }
2102 FUNCTION {music} { booklet }
2103 FUNCTION {video} { booklet }
2104 FUNCTION {dataset} { manual }
2105 FUNCTION {electronic} { manual }
2106 FUNCTION {online} { manual }
2107 FUNCTION {patent} { manual }
2108 FUNCTION {software} { manual }
2109 FUNCTION {standard} { manual }
2110 FUNCTION {www} { manual }
2111 FUNCTION {thesis} { phdthesis }
2112 FUNCTION {report} { techreport }
```

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

```
2113 FUNCTION {default.type} { misc }
```

## 5.15 Bibliography processing

We are now ready for Bib<sub>T</sub>E<sub>X</sub> to read in the `.bib` file.

```
2115 READ
```

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

```
2116 FUNCTION {sortify}
2117 { purify$
2118   "l" change.case$
2119 }
```

`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`.

```
2120 INTEGERS { len }
2121 FUNCTION {chop.word}
2122 { 's :=
2123   'len :=
2124   s #1 len substring$ =
2125     { s len #1 + global.max$ substring$ }
2126   's
2127   if$
2128 }
```

`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.'

```
2129 FUNCTION {format.lab.names}
2130 {'s :=
2131   "" 't :=
2132   #1 'nameptr :=
2133   s num.names$ 'numnames :=
2134   numnames 'namesleft :=
2135     { namesleft #0 > }
2136     { s nameptr
2137       "{vv-}{l1}" format.name$
2138       't :=
2139       nameptr #1 >
2140         { nameptr #2 =
2141           numnames #3 > and
2142             { "others" 't :=
```

```

2143     #1 'namesleft := }
2144     'skip$
2145     if$
2146     namesleft #1 >
2147     { ", " * t * }
2148     { s nameptr "{ll}" format.name$ duplicate$ "others" =
2149       { 't := }
2150       { pop$ }
2151       if$
2152       t "others" =
2153       { " " * bbl.etal *
2154       }
2155       { bbl.and
2156         space.word * t *
2157       }
2158       if$
2159     }
2160     if$
2161   }
2162   't
2163   if$
2164   nameptr #1 + 'nameptr :=
2165   namesleft #1 - 'namesleft :=
2166 }
2167 while$
2168 }

```

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

```

2169 FUNCTION {author.editor.key.label}
2170 { author empty$
2171   { editor empty$
2172     { key empty$
2173       { title empty$
2174         { cite$ #1 #3 substring$ }
2175         { title
2176           type$ "book" =
2177             { emphasize }
2178             'skip$
2179           if$
2180         }
2181       if$
2182     }
2183     'key
2184   if$
2185 }
2186 { editor format.lab.names }
2187 if$
2188 }
2189 { author format.lab.names }
2190 if$
2191 }

```

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

```

2192 FUNCTION {editor.key.label}
2193 { editor empty$
2194   { key empty$
2195     { title empty$
2196       { cite$ #1 #3 substring$ }
2197       { title emphasize }
2198     if$
2199   }
2200   'key
2201 if$
2202 }
2203 { editor format.lab.names }
2204 if$
2205 }

```

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

names.

```
2206 FUNCTION {author.key.label}
2207 { author empty$
2208   { key empty$
2209     { title empty$
2210       { cite$ #1 #3 substring$ }
2211       { title
2212         type$ "article" =
2213         type$ "incollection" =
2214         or
2215         type$ "inproceedings" =
2216         or
2217         type$ "misc" =
2218         or
2219         'skip$
2220         { emphasize }
2221         if$
2222       }
2223     } if$
2224   }
2225   'key
2226   if$
2227 }
2228 { author format.lab.names }
2229 if$
2230 }
```

calc.short.authors chooses between the above.

```
2231 FUNCTION {calc.short.authors}
2232 { type$ "book" =
2233   type$ "inbook" =
2234   or
2235   'author.editor.key.label
2236   { type$ "proceedings" =
2237     'editor.key.label
2238     'author.key.label
2239     if$
2240   }
2241   if$
2242   'short.list :=
2243 }
```

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.

```
2244 FUNCTION {calc.label}
2245 { calc.short.authors
2246   short.list
2247   "("
2248   *
2249   year nodate.check duplicate$ empty$
2250   { pop$ "" }
2251   'skip$
2252   if$
2253   *
2254   'label :=
2255 }
```

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.'

```
2256 FUNCTION {sort.format.names}
2257 { 's :=
2258   #1 'nameptr :=
2259   ""
2260   s num.names$ 'numnames :=
2261   numnames 'namesleft :=
2262   { namesleft #0 > }
2263   { s nameptr
2264     "{ll{ }}{ f{ }}{ jj{ }}{ vv{ }}"
```

```

2265     format.name$ 't :=
2266     nameptr #1 >
2267     { " " *
2268     namesleft #1 =
2269     t "others" =
2270     and
2271     { "zzzzz" 't := }
2272     'skip$
2273     if$
2274     t sortify *
2275     }
2276     { t sortify * }
2277     if$
2278     nameptr #1 + 'nameptr :=
2279     namesleft #1 - 'namesleft :=
2280     }
2281     while$
2282 }

```

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

```

2283 FUNCTION {sort.format.title}
2284 { 't :=
2285   "A " #2
2286   "An " #3
2287   "The " #4 t chop.word
2288   chop.word
2289   chop.word
2290   sortify
2291   #1 global.max$ substring$
2292 }

```

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.

```

2293 FUNCTION {author.editor.sort}
2294 { author empty$
2295   { editor empty$
2296     { title empty$
2297       { key empty$
2298         { "to sort, need author, editor, title, or key in " cite$ * warning$
2299           ""
2300         }
2301         { key sortify }
2302       } if$
2303     }
2304     { title sort.format.title }
2305   } if$
2306 }
2307 { editor sort.format.names }
2308 } if$
2309 }
2310 { author sort.format.names }
2311 } if$
2312 }

```

editor.sort does similar but skips the author names.

```

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

```



```
2327   if$
2328 }
```

author.sort does similar but skips the editor names.

```
2329 FUNCTION {author.sort}
2330 { author empty$
2331   { title empty$
2332     { key empty$
2333       { "to sort, need author, title, or key in " cite$ * warning$
2334         ""
2335       }
2336     { key sortify }
2337     if$
2338   }
2339   { title sort.format.title }
2340   if$
2341 }
2342 { author sort.format.names }
2343 if$
2344 }
```

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.

```
2346 FUNCTION {presort}
2347 { calc.label
2348   label sortify
2349   " "
2350   *
2351   type$ "book" =
2352   type$ "inbook" =
2353   or
2354   'author.editor.sort
2355   { type$ "proceedings" =
2356     'editor.sort
2357     'author.sort
2358     if$
2359   }
2360   if$
2361   #1 entry.max$ substring$
2362   'sort.label :=
2363   sort.label
2364   *
2365   " "
2366   *
2367   title field.or.null
2368   sort.format.title
2369   *
2370   #1 entry.max$ substring$
2371   'sort.key$ :=
2372 }
```

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.

```
2374 ITERATE {presort}
2375 SORT
```

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

```
2376 STRINGS { last.label next.extra }
2377 INTEGERS { last.extra.num last.extra.num.extended last.extra.num.blank number.label }
2378 FUNCTION {initialize.extra.label.stuff}
2379 { #0 int.to.chr$ 'last.label :=
2380   "" 'next.extra :=
2381   #0 'last.extra.num :=
2382   "a" chr.to.int$ #1 - 'last.extra.num.blank :=
2383   last.extra.num.blank 'last.extra.num.extended :=
2384   #0 'number.label :=
2385 }
2386 FUNCTION {forward.pass}
```

```

2387 { last.label label =
2388   { last.extra.num #1 + 'last.extra.num :=
2389     last.extra.num "z" chr.to.int$ >
2390     { "a" chr.to.int$ 'last.extra.num :=
2391       last.extra.num.extended #1 + 'last.extra.num.extended :=
2392     }
2393     'skip$
2394     if$
2395     last.extra.num.extended last.extra.num.blank >
2396     { last.extra.num.extended int.to.chr$
2397       last.extra.num int.to.chr$
2398       * 'extra.label := }
2399     { last.extra.num int.to.chr$ 'extra.label := }
2400     if$
2401   }
2402   { "a" chr.to.int$ 'last.extra.num :=
2403     "" 'extra.label :=
2404     label 'last.label :=
2405   }
2406   if$
2407   number.label #1 + 'number.label :=
2408 }
2409 FUNCTION {reverse.pass}
2410 { next.extra "b" =
2411   { "a" 'extra.label := }
2412   'skip$
2413   if$
2414   extra.label 'next.extra :=
2415   extra.label
2416   duplicate$ empty$
2417   'skip$
2418   { "{\natexlab{" swap$ * "}" * }
2419   if$
2420   'extra.label :=
2421   label extra.label * 'label :=
2422 }

```

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.

```

2423 EXECUTE {initialize.extra.label.stuff}
2424 ITERATE {forward.pass}
2425 REVERSE {reverse.pass}
2426 FUNCTION {bib.sort.order}
2427 { sort.label
2428   " "
2429   *
2430   year field.or.null sortify
2431   *
2432   " "
2433   *
2434   title field.or.null
2435   sort.format.title
2436   *
2437   #1 entry.max$ substring$
2438   'sort.key$ :=
2439 }
2440 ITERATE {bib.sort.order}
2441 SORT

```

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

```

2442 FUNCTION {begin.bib}
2443 { preamble$ empty$
2444   'skip$
2445   { preamble$ write$ newline$ }
2446   if$
2447   "\begin{thebibliography}{ number.label int.to.str$ * "}" *
2448   write$ newline$
2449   "\providecommand{\natexlab}[1]{#1}"
2450   write$ newline$

```

```

2451 "\providecommand{\url}[1]{\texttt{#1}}"
2452 write$ newline$
2453 "\providecommand{\urlprefix}{Available from: }"
2454 write$ newline$
2455 "\providecommand{\urldateprefix}{Accessed }"
2456 write$ newline$
2457 "\providecommand{\selectlanguage}[1]{\relax}"
2458 write$ newline$
2459 "\providecommand{\bibinfo}[2]{#2}"
2460 write$ newline$
2461 "\providecommand{\eprint}[2][\url{#2}]"
2462 write$ newline$
2463 }
2464 EXECUTE {begin.bib}
2465 EXECUTE {init.state.consts}

```

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

```

2466 ITERATE {call.type$}

```

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

```

2467 FUNCTION {end.bib}
2468 { newline$
2469 "\end{thebibliography}" write$ newline$
2470 }
2471 EXECUTE {end.bib}

```