The `urlbst` package

Norman Gray
<http://nxg.me.uk>

Version 0.7, 2011 July 20

The `urlbst` package consists of a Perl script which edits BibTEX style files (`.bst`) to add a webpage entry type, and which adds a few new fields – notably including url – to all other entry types. The distribution includes preconverted versions of the four standard BibTEX `.bst` style files.

It has a different goal from Patrick Daly's `custom-bib` package [**?**] – that is intended to create a BibTEX style `.bst` file from scratch, and supports url and eprint fields. This package, on the other hand, is intended for the case where you already have a style file that works (or at least, which you cannot or will not change), and edits it to add the new webpage entry type, plus the new fields.

The added fields are:

- url and lastchecked, to associate a URL with a reference, along with the date at which the URL was last checked to exist;
- doi, for a reference's DOI (see `http://doi.org`);
- eprint, for an arXiv eprint reference (see `http://arxiv.org`); and
- pubmed for a reference's PubMed identifier (PMID, see `http://pubmed.gov`).

Usage is simple:

```
% urlbst [--eprint] [--doi] [--pubmed]
    [--nohyperlinks] [--inlinelinks] [--hypertex] [--hyperref]
    [input-file [output-file]]
```

where the `input-file` is an existing `.bst` file, and the `output-file` is the name of the new style file to be created. If either file name is missing, the default is the standard input or standard output respectively. For example:

```
% urlbst --eprint bibstyle.bst
```

would convert the style file `bibstyle.bst`, including support for e-prints, and sending the result to the standard output (ie, the screen, so it would more useful if you were to either redirect this to a file or supply the output-file argument).

If the option `--eprint` is present, then we switch on support for `eprint` fields in the modified .bst file, with a citation format matching that recommended in `http://arxiv.org/help/faq/references`. If the option `--doi` is present, we include support for a `doi` field, referring to a Digital Object Identifier (DOI) as standardised by `http://www.doi.org/`. And if `--pubmed` is present, we include

support for a `pubmed` field, referring to a PubMed identifier as supported at `http://www.pubmed.gov`.

If either of the `--hypertex` or `--hyperref` options is present, then the generated .bst file includes support for hyperlinks in the generated eprint entries in the bibliography, with the format being either HyperTEX (see `http://arxiv.org/hypertex/#implementation`), supported by xdvi, dvips and others, or using the support available from the `hyperref` package. These options have no effect unless one of the `--eprint`, `--doi`, `--pubmed` or `--inlinelinks` options is given: when URLs are included in the bibliography, they are written out using the `\url{...}` command. The `hyperref` support is more generic, and more generally supported, and so you should choose this unless you have a particular need for the HyperTEX support. The `--nohyperlinks` option, which is present by default, suppresses all hyperlinking.

By default, any URL field is displayed as part of the bibliography entry, linked to the corresponding URL via the mechanism managed by the `--hypertex` and `--hyperref` options. If the `--inlinelinks` option is present, however, then the URL is not displayed in the printed entry, but instead a hyperlink is created, linked to suitable text within the bibliography entry, such as the citation title. This option does not affect the display of eprints, DOI or PubMed fields. It makes no sense to specify `--inlinelinks` with `--nohyperlinks`, and the script warns you if you do that, for example by failing to specify one of the link-style options. This option is (primarily) useful if you're preparing a version of a document which will be read on-screen; the point of it is that the resulting bibliography is substantially more compact than it would otherwise be.

The support for all the above behaviours is always included in the output file. The options instead control only whether the various behaviours are enabled or disabled, and if you need to alter these, you may do so by editing the generated `.bst` file and adjusting values in the `{init.urlbst.variables}` function, where indicated.

The generated references have URLs inside `\url{...}`. The best way to format this this is with the `url` package (see [**?**] for pointers), but as a poor alternative, you can try `\newcommand{\url}[1]{\texttt{#1}}`. The `hyperref` package automatically processes `\url{...}` in the correct way to include a hyperlink, and if you have selected `hyperref` output, then nothing more need be done. If you selected HyperTEX output, however, then the script encloses the `\url` command in an appropriate HyperTEX special.

When the style file generates a link for DOIs, it does so by prepending the string `http://dx.doi.org/` to the DOI. This is generally reasonable, but some DOIs have characters in them which are illegal in URLs, with the result that the resulting `dx.doi.org` URL doesn't work. The only real way of resolving this is to write a URL-encoding function in the style-file implementation language, but while that would doubtless be feasible in principle, it would be hard and very, very, ugly. The only advice I can offer in this case is to rely on the fact that the DOI will still appear in the typeset bibliography, and that users who would want to take advantage of the DOI will frequently (or usually?) know how to resolve the DOI when then need to. As a workaround, you could include a URL-encoded DOI URL in the `url` field of the entry (thanks to Eric Chamberland for this suggestion).

The `urlbst` script works by spotting patterns and characteristic function names in the input `.bst` file. It works as-is in the case of the four standard

```
@Manual{w3chome,
  url =          {http://www.w3.org},
  title =        {The World Wide Web Consortium},
  year =         2009,
  lastchecked =  {26 August 2009}}

@Book{schutz,
  author =      {Bernard Schutz},
  title =       {Gravity from the GroundUp},
  publisher =   {Cambridge University Press},
  year =        {2003},
  url =         {http://www.gravityfromthegroundup.org/},
  lastchecked = {2008 June 16}}
```

Figure 1: The new `@webpage` entry type, and the `url` field in action

BibTEX style files `plain.bst`, `unsrt.bst`, `alpha.bst` and `abbrv.bst`. It also works straightforwardly for many other style files – since many of these are derived from, or at least closely inspired by, the standard ones – but it does not pretend that it can do so for all of them. In some cases, such as the style files for the `refer` or `koma-script` packages, the style files are not intended to be used for formatting; others are sufficiently different from the standard files that a meaningful edit becomes impossible. For the large remainder, however, the result of this script should need only relatively minor edits before being perfectly usable.

New `.bib` entry and field types

The new entry type webpage has required fields title and url, and optional fields author, editor, note, year, month and lastchecked. The url and lastchecked fields are new, and are valid in other entry types as well: the first, obviously, is the URL which is being cited, or which is being quoted as an auxiliary source for an article perhaps; the second is the date when you last checked that the URL was there, in the state you quoted it; this is necessary since some people, heedless of the archival importance of preserving the validity of URLs, indulge in the vicious practice of reorganising web pages and destroying links. For the case of the webpage entry type, the editor field should be used for the 'maintainer' of a web page.

For example, in Figure 1 we illustrate two potential `.bib` file entries. The `@webpage` entry type is the new type provided by this package, and provides reference information for a webpage; it includes the new `url` and `lastchecked` fields. There is also an example of the standard `@book` entry type, which now includes the `url` and `lastchecked` fields as well. The difference between the two references is that in the `@book` case it is the book being cited, so that the `url` provides extra information; in the `@webpage` case it is the page itself which is of interest. You use the new `eprint`, `doi` and `pubmed` fields similarly, if the bibliographic item in question has an e-print, DOI or PubMed reference.

How do you use this in a document? To use the the `alphaurl.bst` style – which is a pre-converted version of the standard `alpha.bst` style, included

in the `urlbst` distribution – you simply make sure that `alphaurl.bst` is in your BibTeX search path (use the command `kpsepath bst` to find this path and `kpsewhich alphaurl.bst` to confirm that BibTeX can find it) and add `\bibliographystyle{alphaurl}` to your LaTeX document.

Sources

There are various sources which suggest how to format references to web pages. I have followed none specifically, but fortunately they do not appear to materially disagree.

ISO-690 [**?**] is a formal standard for this stuff. Walker and Taylor's *Columbia Guide to Online Style* [**?**] provides extensive coverage (but is only available on dead trees). There are two style guides associated with the APA, namely the published APA style guide [**?**] (a paper-only publication, so should be ignored by all, if there's any justice in the world), and what appears to be the 1998 web-citation proposal for that [**?**], which also includes some useful links. The TeX FAQ [**?**] has both practical advice and pointers to other sources.[1]

Hints

If you use Emacs' BibTeX mode, you can insert the following in your `.emacs` file to add knowledge of the new webpage entry:

```
(defun my-bibtex-hook ()
  (setq bibtex-mode-user-optional-fields '("location" "issn")) ; e.g.
  (setq bibtex-entry-field-alist
        (cons
         '("Webpage"
           ((("url" "The URL of the page")
             ("title" "The title of the resource"))
            (("author" "The author of the webpage")
             ("editor" "The editor/maintainer of the webpage")
             ("year" "Year of publication of the page")
             ("month" "Month of publication of the page")
             ("lastchecked" "Date when you last verified the page was there")
             ("note" "Remarks to be put at the end of the entry"))))
         bibtex-entry-field-alist)))
(add-hook 'bibtex-mode-hook 'my-bibtex-hook)
```

After that, you can add a webpage entry by typing `C-c␣C-b␣webpage` (or `M-x␣bibtex-entry`).

It is a *very* good idea to use the `url` package: it deals with the problem of line-breaking long URLs, and with the problem that BibTeX creates, of occasionally inserting %-signs into URLs in generated bibliographies.

See also the URL entry in the UK TeX FAQ [**?**], and references therein.

Acknowledgements: Thanks to Jason Eisner for suggesting the `--inlinelinks` option, to Paweł Widera for the suggestion to use `\path` when formatting DOIs and to Michael Giffin for the suggestion to include PubMed URLs.

[1] Emory University's Goizueta Business Library once had a collection of useful links on this topic, but they've whimsically changed the URL at least twice since I first distributed `urlbst`, and I've got fed up fixing their broken link.