The New (LaTeX 2_ε) *TUGboat* Macros

Robin Fairbairns

1 Introduction

This note reports on a new set of macros for use by *TUGboat* authors. The macros represent a development of the earlier ltugboat and ltugproc styles that were written for use with LaTeX 2.09. The development was started by others than I, notably Sebastian Rahtz, Michel Goossens, Nico Poppelier and Johannes Braams; I have been assisted in my work both directly and indirectly by many others, including Barbara Beeton, Mimi Burbank and (of course) the LaTeX3 team.

The ultimate aim of the work is to provide a single production environment for the whole of *TUGboat*. That aim has not (yet) been achieved, though it is still believed to be achievable for a significant proportion of papers.[1]

2 Availability

The macros are released for general use, and are distributed via CTAN[2] in the usual LaTeX way as files `tugboat.dtx` and `tugboat.ins`. When the `.ins` file is processed by LaTeX, the files `ltugboat.cls`, `ltugproc.cls` and `ltugbib.bst` (for ordinary production work) and `ltugcomn.sty` (a cooking pot of useful macros, for documentation, etc.) are produced. The `.dtx` file may itself be processed by LaTeX to produce a formatted (somewhat 'literate') source listing for those who feel they need more detailed description of the macros than is offered in the present note.

3 The general structure of a paper

The author need not understand the details of the production of *TUGboat* — her business is to produce the text that the editor is to mould into the body of an issue.

However, any modern author wants to be able to view the progress of her efforts; for this, she needs to use the *TUGboat* class `ltugboat.cls` — it defines the way an article in a normal issue of *TUGboat* will appear, and is therefore a useful aid.

Similarly, one submitting a paper for presentation at an annual TUG conference may use the `ltugproc.cls`, which will set the paper as it should appear in a proceedings issue of *TUGboat*.

[1] *TUGboat* will continue to accept papers written using the 'plain' macros; it's probably fanciful to hope that *every* paper can be moulded into use within LaTeX.

[2] From directory `macros/latex/contrib/supported/tugboat`

Each paper, therefore, is written as a document that may stand on its own. It starts with a `\documentclass` command, and its body is enclosed in a document environment. Each of the document classes specifies a set of options, described in the next two subsections (3.1 and 3.2). In the ordinary course of events the author needn't bother with any of these options, since the defaults are designed for creating proof copies of papers.

Since the author typesets her paper in a 'proof mode', the paper's published appearance will differ from its appearance at the time of submission. The changes required in this process are the responsibility of the *TUGboat* production team, and the author need not be concerned with them.

3.1 Class options: the ltugboat class

The ltugboat class accepts all the options of the article class (though it suppresses the font-size selection, one/two-side and one/two-column options).

draft Set up for a draft copy of a paper (this is the default setting — the author need not explicitly set it): page numbering to start at 1001, black marks for overfull boxes, and no registration marks.

extralabel Use the extra label-distinguishing mark in the body of the reference; see section 11.

final Set up for the final copy of a paper: page numbering to come from elsewhere, no black marks, and registration marks to be used.

harvardcite Specify Harvard-style citation; see section 11.

noextralabel Don't use the extra label-distinguishing mark in the body of the reference; see section 11.

nonumber Sections are not to be numbered; section heading layout is to be as in the 'plain' tugboat styles.

numbersec Sections, subsections and subsubsections are to be numbered (this is the default setting — the author need not explicitly set it).

preprint Set up for a preprint.

rawcite Specify default (numbered) citation; see section 11.

3.2 Class options: the ltugproc class

The ltugproc class accepts all of the options allowed with ltugboat. In addition, it accepts an option that specifies the year of the conference for which the paper is submitted. For example, this year's conference would be specified by option tug98.

The reason that such an option is necessary is that the type design of papers for each year's proceedings issue is decided (in detail) by the editor.

preliminary draft, February 20, 1998 11:05

By default, ltugproc class selects the option for this year's conference. The class reads a configuration file `ltugproc.cfg`, in which you may specify the year by a command of the form:

```
\newcommand{\tugProcYear}{97}
```

The class deals with two-digit year selections for conferences up to 2069 (which date was chosen entirely at random, uninfluenced, of course, by the Unix™Epoch...).

4 Command syntax

We would have liked to offer perfectly uniform syntax for people to use when preparing their papers. Unfortunately, uniform syntax is not available with any widely-available set of macros (though see, for example, the discussions in **?**, **?**, and **?**). In the circumstances, we have sought simply to keep to the spirit of Lamport's (**?**), as modified[3] by the LaTeX 2_ε work (see, for example, **?**).

In the few cases that it has proved possible to emulate (what seems to a staid old LaTeX programmer, such as the present author) the gay abandon of the syntax of the 'plain' tugboat styles (**?**), we have done. Nevertheless, on the whole, the new ltugboat macros simply define LaTeX commands and environments, or modify the definitions of LaTeX 'standard' commands. Section 12 lists equivalences between macros defined by the 'plain' package and those defined by the new package.

The 'down' side of this decision is, of course, the 'welter of LaTeX braces' that Barbara Beeton has been heard to complain of in production team discussions. One has to hope that the (near) uniformity of syntax offers those who think like Barbara some small recompense!

5 Divisions of the paper

Papers in *TUGboat* may be subdivided in the normal way of a LaTeX article (the classes are defined in terms of LaTeX's article class). Thus the author may use `\section`, `\subsection`, ..., `\paragraph` commands (but `\part` and `\subparagraph` from article are suppressed, and `\chapter`, which doesn't even appear in the parent class, receives the same treatment).

Authors should note that the style of ordinary issues of *TUGboat* makes no distinction between the titles of the divisions; the visual style relies on the section numbers to indicate where the divisions lie in the hierarchy. As a result, the un-numbered '`*`' forms of the `\section`, etc., commands, are inappropriate. The ltugboat class therefore warns the author who (possibly inadvertently) uses one of these forms. (The author who wishes to use un-numbered sections throughout her paper may use the nonumber class option (see 3.1).

By contrast, the conference proceedings issues of *TUGboat* do *not* number their sections, by default. As a result the use of '`*`' forms is equally inappropriate in ltugproc class, and the class equally warns the author. Since the sections don't have numbers, references to section numbers don't work; they therefore (once they're resolved) warn the author of potential problems.

Reference may, however, be made to the 'title' of divisions of the paper, whether they are numbered or not. The `\nameref` command (which uses the technique developed for the hyperref package (**?**)) permits such references; for example, the present section was introduced by:

```
\section{Divisions of the paper}
\label{sec:divs-paper}
```

and the command `\nameref{sec:divs-paper}` produces 'Divisions of the paper'.

5.1 Abstracts

The classes make provision for abstracts, but the provision is different for the two classes.

The ltugboat class provides two environments, abstract and longabstract. The abstract environment simply typesets its body as an un-numbered section whose title is 'Abstract'. The longabstract environment typesets its body in small text, and separates the abstract from the rest of the paper with a decorative line.

The ltugproc class typesets abstracts as part of the title of the paper, below the names and addresses of the authors, but before the main body of the paper. Therefore, for proceedings articles, the abstract environment must appear in the source of the paper *before* the `\maketitle` command (it may even appear before `\begin{document}`, if the author so chooses).

5.2 Appendices

A paper may have appendices, which are expressed in exactly the same way as they would be in LaTeX article class:

```
\appendix
\section{This is appendix A}
...
\section{This is appendix B}
```

Which will produce 'section' headings similar to:

A This is appendix A

[3] The cited edition of Lamport's book documents the modified version of LaTeX, but it's worth emphasising that we use the modified version as the reference.

TUGboat articles may have a small extension to this format: this extension was originally developed for proceedings issues, but is also available in normal issues:

```
\begin{appendix}
\section{This is the first one}
...
\end{appendix}
```

Which will produce 'section' headings similar to:

Appendix A This is the first one

In both cases, the subsections are numbered as normal (i.e., as 'A.n' in normal *TUGboat* papers, and not at all in proceedings classes).

6 Titles, addresses and so on

The title and author(s) of a paper are quoted using commands that are familiar (in syntax, at least) to most LaTeX users; the `\title` command is exactly that used in the standard LaTeX classes.

The `\author` command is used once for each co-author of the paper, and for each `\author` there should be a `\address` command that gives a (postal) correspondence address. In addition (wherever possible), *TUGboat* likes to quote an email address for authors: for this, the `\netaddress` command is used. Finally, each author may advertise a 'home' Web page, using a `\personalURL` command.

For example, the present paper has at its start:

```
\title{The New (\LaTeXe) \TUB{} Macros}
\author{Robin Fairbairns}
\address{University of Cambridge
    Computer Laboratory\\
  Pembroke Street,\\
  Cambridge, CB2 3QG,\\
  UK}
\netaddress{rf@cl.cam.ac.uk}
\personalURL{http://www.cl.cam.ac.uk/users/...}
\maketitle
```

Since this paper is prepared for a normal issue of *TUGboat* and therefore uses the ltugboat class, the `\maketitle` merely typesets the title of the paper and the author's name. The address(es) are typeset at the end of the paper where the author gives a `\makesignature` command.

The ltugproc class typesets all the information about the author at the head of the paper, when the `\maketitle` is given; this class disallows the `\makesignature` command.

Note that the author had a problem typesetting the above example verbatim: the lines are too long. If the information being given is to be typeset as ordinary text (as in the case of the `\address` line above), it can be 'wrapped' perfectly happily, as in normal text. If one of the verbatim items (`\netaddress` or `\personalURL` commands) is going to be too wide for the column, what is the author to do? (Abbreviating the text, as in the `\personalURL` above, is *not* usually an acceptable option!) Unfortunately, the `%` sign is an entirely acceptable element of both email addresses and URLs, so that the normal 'fall-back' isn't available. Therefore, the classes typeset these electronic addresses in an environment where some of the characters (notably '`.`' and '`/`') are treated as word-divisions for the purposes of laying out the line. This is visible in the case of my personal URL, in the signature block at the end of the present paper.

If the paper is the result of more than one author's labours, a sequence of `\author`, `\address`, `\netaddress` and `\personalURL` commands may be given, as in the following, which comes from a paper given at TUG'95 (slightly edited):

```
\author{Michel Goossens}
\address{CN Division, CERN\\
  ...}
\netaddress{...}

\author{Sebastian Rahtz}
\address{Elsevier Science Ltd\\
  ...}
\netaddress{...}

\author{Robin Fairbairns}
\address{University of Cambridge
    Computer Laboratory\\
  ...}
\netaddress{...}
\personalURL{...}
```

The class files will take care of arranging author names and addresses between the `\maketitle` and (possibly) `\makesignature` commands.

6.1 Compilation articles

Compilation articles are built from a set of contributed parts, and are under the general (sub-) editorship of the author[4] of the article. The author of the article is presented (using `\author` and suchlike commands) in the usual way, and writes the introductory text. Each contributors' part then follows. The contributor's name is quoted in the `\contributor` command, which is an analogue of the `\author` command; contributors' `\address`, `\netaddress` or `\personalURL`. The `\contributor` command opens a group in which the contribution appears, and the contributor's signature (produced

[4] Or authors: there's no reason in particular that compilation articles should not be put together more than one person

with a \makesignature command) closes the group. The general scheme looks like:

```
\title{Example compilation article}
\author{Robin Fairbairns}
\address{University of Cambridge ...}
\netaddress{...}
 ...
 introductory text
 ...
\makesignature

\contributor{Betsy the Dog}
\address{Romsey Town, Cambridge}
 ...
 Betsy's contribution
 ...
\makesignature

...
```

7 Verbatim text

The classes do not at present provide the same wide range of facilities as the 'plain' tugboat style (?); the author had hoped to 'borrow' facilities from a package which is believed to be in development, but in the event that package has not materialised.

For in-line verbatim text, authors should ordinarily employ the facilities of LaTeX itself (the \verb macro). This macro, of course, is highly restricted as to its usage (primarily, that it may not appear in the argument of *any* other macro, even \footnote).

For 'display verbatim' (to employ the term used by ?), the classes add a small increment to the functionality of LaTeX's verbatim environment, by introducing an optional argument. The optional argument may contain commands to be executed before starting the verbatim text; the set of commands which have useful effect is strictly limited, but the following are commonly used:

- Font size selection commands: for example, all the display verbatim in the present paper starts with:

  `\begin{verbatim}[\small]`
- The command \ruled, which is available *only* in verbatim's optional argument, and specifies that a column-wide rule should be drawn before and after the verbatim text
- One of the \make* commands,[5] which change the category code of characters within the verbatim text. This is (of course) a facility that should only be used with the utmost caution, but it can, for example, be employed to provide interesting effects by knowledgeable authors.

[5] \makeescape, \makebgroup, ..., \makecomment; used, for example, as \makeescape\|

Two caveats about the use of this facility should be noted:

- The search for the optional argument can be confused by the appearance of a [character as the first of the displayed verbatim. An author who wishes to start verbatim text with a [character should provide an empty optional argument (i.e., simply '[]') to the verbatim environment.
- The facility is lost when certain packages are loaded. An example is the verbatim package (?), which redefines the verbatim environment in its entirety. Of course, any package that loads verbatim (such as moreverb, ?) will necessarily have the same effect. (It should be noted that verbatim and moreverb provide some of the facilities that are available in the 'plain' tugboat styles, so that an author could reasonably be tempted to use them. There is no objection in principle to authors using these packages.)

8 Floating inserts

The classes do not make any change to LaTeX's built-in provision for floating inserts, so that authors may generate figures and tables just as they would in any 'normal' LaTeX document. Figure and table captions, and labels referring to them, are also substantially untouched.

However, since both classes typeset in two columns, authors must distinguish between the figure and table environments (which produce floats that are the same width as the column) and the figure* and table* (which produce floats that are the same width as the page).

9 Special-purpose typesetting

The classes define a rather large set of commands for special-purpose typesetting. Some of them are available for historical reasons only, and many are only useful in somewhat restricted circumstances. For this reason, the present paper only outlines a representative, small set of the macros.

9.1 Acronyms and logos

The classes provide macros that produce 'correct' representations of a large number of acronyms and logos; a small representative selection is shown in figure 1.

Authors are urged to note the \acro command, which is defined in the classes. The visual appearance of (mostly) lower-case English text, with interpolated acronyms in the same point size, is generally unpleasing. Therefore, the \acro command

Macro	*Output*
`\CTAN`	CTAN
`\eTex`	ε-TEX
`\HTML`	HTML
`\ISBN`	ISBN
`\ISSN`	ISSN
`\MF`	METAFONT
`\MP`	METAPOST
`\NTS`	$\mathcal{NTS}$
`\OMEGA`	Ω
`\OTP`	ΩTP
`\SGML`	SGML
`\TUB`	*TUGboat*
`\TUG`	TEX Users Group
`\tug`	TUG

Figure 1: Some of the classes' acronyms and logos

typesets its argument slightly smaller than it would otherwise appear: compare 'URL' (`\acro{URL}`, as used above) with 'URL'. Many macros that simply generate calls to `\acro` are defined by the classes; two examples, `\CTAN` and `\tug` of the list in figure 1 have already been used in the present paper.

9.2 Other special typesetting

A small list of special typesetting commands follows: a large set of such commands is defined in the classes, but the list covers most of the 'everyday' ones.

`\cs{`*cmd*`}` Typeset a control sequence name. (The command `\cs{fred}` is typeset as `\fred`.)

`\env{`*environment*`}` Typeset an environment name. (The command `\env{fred}` is typeset as `\begin{fred}`.)

`\meta{`*arg*`}` Typeset a formal argument name. (The command —⟨*fred*⟩— is typeset as ⟨*fred*⟩.)

`\Dash` Typeset an em-dash, surrounded by thin spaces, only breakable *after* the dash; this is the preferred method of specifying a dash in running text.

`\dash` Typeset an en-dash, in the same way as `\Dash` does.

`\nth{`*n*`}` Typeset an ordinal number. For example, `\nth{1}` is set as 1$^{\text{st}}$, `\nth{27}` is set as 27$^{\text{th}}$, and so on.

`\sfrac{`*num*`}{`*denom*`}` Typeset a fraction to match running text; for example `\sfrac{3}{4}` is set as $^3/_4$.

10 Use of packages

In general, the *TUGboat* team will be sympathetic to authors who wish to use non-standard packages in their papers; indeed, in a journal devoted to the usage of TEX, the editor would be churlish indeed to refuse such usage. However, the team does need to be able to process the paper on the *TUGboat* production computers, and the author who is slapdash about the way she submits her paper will cause confusion and delay. (Remember that the editorial team is a group of volunteers, each working in his or her spare time.)

In general, packages currently on CTAN, and known to work with *current* LATEX,[6] are unlikely to give problems.

In particular, the team is happy to accept papers using packages that are supported by members of the LATEX3 team,[7] subject to the two provisos:

- Use of the verbatim package has implications for the verbatim facilities provided by the classes — see section 7.
- Use of babel almost inevitably implies use of hyphenation patterns that the team may not have installed in their LATEX format; it is therefore important that the author explains her babel configuration to the editorial team. The minimum documentation required is a copy of the author's `language.dat` file, and copies (or CTAN pointers to) any hyphenation files used.

Usage of other packages should always be subject to negotiation with the team. If the team does not have access to a copy of the package, life is going to be very difficult; authors are urged to be sensible in this regard. A sensible mechanism for submitting out-of-the-ordinary packages (as for paper-specific bibliographies) is by use of the filecontents environment.

TUG has a policy that macro packages described in *TUGboat* should be available for readers to use. Since typing macros from printed sources is such an error-prone undertaking, authors of publicly available packages are urged to submit their macros to the CTAN archives. If a package is only available under restricted terms, authors are urged to make this fact clear when first submitting an article to the editor.

Some facilities are considered inappropriate to delivery by the *TUGboat* classes, and as a result, the *TUGboat* team recommend certain packages to authors.

At present, the list of recommended packages consists of only two, `mflogo.sty` (?) and `url.sty` (?).

[6] The team's version of LATEX is regularly updated, though not during the course of a production run.

[7] Those in the LATEX base distribution, or one of those on the `macros/latex/packages` sub-tree on CTAN.

Both classes will load the mflogo package if it is present on the author's system; if the package is not present, the classes will emulate its more important features; the package defines METAFONT and METAPOST logos using recent versions of Knuth's `logo10` font family.

The url package is useful when one is typesetting significant numbers of file names, network addresses or URLs; it is being used in the present paper (not least in the bibliography).

11 Bibliography

Bibliographic citations give much grief to the editorial team. Good publishing practice requires that there be editorial control of the way citations in a journal are presented yet, all too often, authors submit articles whose bibliography is formatted according to their preference. The important rules for authors, then, are that they *shouldn't* supply a `.bbl` file (BibTeX processed output), and that they *shouldn't* write out a thebibliography environment, but should rather submit a working `.bib` file with their paper.[8] As with uncommon packages, the filecontents environment is a convenient way to deliver the bibliography file.

A special case is made of the accumulated bibliography of *TUGboat* itself;[9] it is always available to the production team, so that authors may make reference to items from that `.bib` file without further ado.

Two citation styles are supported within *TUGboat* articles, '*raw*' and '*harvard*' (the present article is employing *harvard* citation). The raw citation style uses the 'standard' BibTeX 'plain' (numeric) citation style; its modification by use of Donald Arseneau's cite package[10] is acceptable. Raw citation is selected by default (by execution of class option rawcite).

Harvard citation may be selected by specifying harvardcite as an option of the `\documentclass` command. The macros used derive pretty directly from the *harvard* styles written by Glenn Paulley and now maintained by Peter Williams; the BibTeX style derives from one developed by Patrick Daly.

[8] The program bibextract, available from CTAN in directory `biblio/bibtex/utils/bibextract`, provides a convenient mechanism for extracting just the relevant portions of your bibliography database for submitting your paper; many of the common BibTeX-database management packages, such as BibDB or BibTools offer similar facilities.

[9] Available on CTAN as `digests/tugboat/biblio/tugboat.bib`

[10] Available on CTAN as `macros/latex/contrib/supported/cite/cite.sty`

The basic citation format is 'author-year', but the macros are capable of many variations: this in turn places somewhat of a load on the author to use the correct citation macro. The macros available are shown in figure 2; the figure assumes an entry in the bibliography with authors Tom, Dick, and Harry, and with a 1990 date.

Macro	*Output*
`\cite{key}`	(Tom, Dick, and Harry, 1990)
`\citeA{key}`	(Tom, Dick, and Harry)
`\citeNP{key}`	Tom, Dick, and Harry, 1990
`\citeANP{key}`	Tom, Dick, and Harry
`\citeN{key}`	Tom, Dick, and Harry (1990)
`\shortcite{key}`	(Tom et al., 1990)
	[also has `A` and `NP` variants]
`\citeyear{key}`	(1990)
	[also has an `NP` variant]

Figure 2: The range of citations in harvard style

Note that, if Tom, Dick, and Harry are a prolific team, there can easily be more than one reference to their work in one year. In such a case, the citations will be (Tom, Dick, and Harry, 1990a), (Tom, Dick, and Harry, 1990b), and so on. These extra 'a', 'b', etc., tags may also appear in the references section of the paper, attached to the year recorded for the reference: whether this indeed happens is controlled by the extralabel and noextralabel class options. The default state (option extralabel) attaches the extra characters.

12 Equivalences between the 'plain' and the LaTeX packages

A good proportion of the commands in the 'plain' packages also appear (with the same meaning) in the LaTeX classes. Figure 3 gives a brief summary of where the macros differ significantly.

Plain macro	LaTeX macro
`\head`	`\section`
`\subhead`	`\subsection`
`\subsubhead`	`\subsubsection`
`\list`	itemize, enumerate, etc., environments
`\verbatim`	verbatim or `\verb`
`\figure`	figure or figure* environments

Figure 3: Equivalences between plain and LaTeX macros

LaTeX itself makes comprehensive provision for lists; the *TUGboat* classes make no attempt to emulate the list facilities of the 'plain' macros.

The 'plain' styles' provision for verbatim text is also somewhat different from the LaTeX approach; the *TUGboat* classes offer a small subset of the extra facilities that the 'plain' styles provide; for more elaborate facilities, the user is referred to the verbatim and moreverb packages (see section 7).

Of course, the syntax of commands given to the LaTeX classes is different (as discussed in section 4); arguments are (almost always) enclosed in braces, and neither of the forms of argument provision promulgated by the 'plain' macros (`\macro`⟨*argument*⟩ `\endmacro` and `\macro *` ⟨*argument*⟩ `*`) are provided by the LaTeX classes.

◇ Robin Fairbairns
University of Cambridge Computer Laboratory
Pembroke Street,
Cambridge CB2 3QG,
UK
`rf@cl.cam.ac.uk`
URL: `http://www.cl.cam.ac.uk/users/rf/robin.html`