

# UNLEASHING THE AWESOME POWER OF L<sup>A</sup>T<sub>E</sub>X

SAM COSKEY

## CONTENTS

1. Introduction	1
2. Using L <sup>A</sup> T <sub>E</sub> X in the Linux Environment	1
2.1. Compiling stuff	1
2.2. Editing your document	3
2.3. A note on X	3
2.4. Figures with xfig	4
2.5. Submissions with tar	4
3. A New Environment	4
4. Customizing Common Commands	5
5. A Figure with Nice Fonts	5
6. Using the A <sub>M</sub> S stuff	6
7. Bibliographies with BibT <sub>E</sub> X	6
References	7

## 1. INTRODUCTION

The following are established facts.

- For any task requiring mathematical typesetting, L<sup>A</sup>T<sub>E</sub>X is the ultimate tool.
- L<sup>A</sup>T<sub>E</sub>X is a pain in the ass to use.

From these two facts we may reason

- Unfortunately, for the best results we must continue to use L<sup>A</sup>T<sub>E</sub>X.
- It is advisable to steps to reduce the magnitude of headache one gets when doing so.

If your head doesn't hurt when you use L<sup>A</sup>T<sub>E</sub>X, then this paragraph can safely be the last that you read from this document. You can't go too wrong with PCTeX or whatever it is that you use. Just don't use bitmapped figures.

## 2. USING L<sup>A</sup>T<sub>E</sub>X IN THE LINUX ENVIRONMENT

**2.1. Compiling stuff.** There are just a few commands you'll need to know to get off the ground. Lets suppose temporarily that your document source file is called `report.tex` (though i don't recommend that you actually use this name). To process your file initially, you simply type

```
latex report
```

When you do so, `latex` will probably spit a lot of errors at you, most of them non-fatal. If it asks you for input with a `?` symbol, just note the line number and the problem it reports and hit return. If it asks you for input more than 7 times, hit CTRL-D and fix some of the errors. If you have references, you may need to run `latex` twice, as indicated by the terminal output. If everything went well, `latex` will produce the file `report.dvi`, which you can view by typing

```
xdvi report &
```

If you later change your source and recompile using `latex`, there is no need to run `xdvi` again. Just click in the existing `xdvi` window to see the updated version.

The above process works pretty well while you're writing your document. One typically edits their sourcefile (discussed next), compiling every 10 minutes or so to see what it looks like.

But however you create your document, you need to totally change gears when it comes time to print it out. You will have to configure the printer correctly at least once. This is a process which changes from year to year, and you may be at least partially on your own. For example, this year I needed to add the line

```
setenv PRINTER cmu-b022@entropy
setenv LPDEST cmu-b022@entropy
```

to my `.cshrc` file to get `lpr` to print to the MSCC printer. I also had to type

```
texconfig paper letter
```

to get the margins right.

The printer printing print command is `lpr`, and this can be applied either to plain text or postscript files. To print your document, though, you can just use this simple command

```
dvips report
```

Once you have done this, you can check the status of the printer queue and the number assigned to your printing job with `lpq`, and you can cancel the job with `lprm`. If you don't want to send your document to the printer, but rather to a file by the name `report.ps`, just type

```
dvips report -o
```

instead. You can view the postscript file with

```
gv report &
```

and you can print it directly with

```
lpr report.ps
```

The last thing you might want to do is distill your document into a `.pdf` file. It is recommended that you follow this recipe

```
dvips -Ppdf report
ps2pdf report.ps
```

Information on all of the commands listed here can be found in the unix manual pages, accessed with

```
man commandname
```

For information on the installation of `TEX` which I suspect is on your system, type

```
info web2c
```

For much more information on the same topic, see [9].

**2.2. Editing your document.** But before you can actually write a L<sup>A</sup>T<sub>E</sub>X document, you need to know how to use a popular unix editor, such as `emacs`, `vim`, etc. I use `emacs`, so I'll give enough basic commands for you to get going. Please note that a very useful tutorial on `emacs` can be found by typing

```
C-h t
```

in an `emacs` window. REF.

Before you even get started with `emacs`, it is recommended that you add the following lines to your `.emacs` file

```
(setq tex-dvi-view-command "xdvi")
(global-font-lock-mode t)
(show-paren-mode t)
```

Now, invoke `emacs` in the usual way.

```
emacs report.tex &
```

If given a file with `.tex` at the end, `emacs` will enter T<sub>E</sub>X-mode, a nice and syntax-highlighted way of editing L<sup>A</sup>T<sub>E</sub>X. Many of the actions you'd like to do in `emacs` (such as saving) can be found in the toolbar menus; their keyboard shortcuts can also be found there. General help is found in the menus.

You should learn the basic navigation shortcuts like `C-a` and `C-e`, editing shortcuts like `C-k` and `C-y`, as well as searching with `C-s` and `C-r` and find/replace with `M-%`.

You can get specific T<sub>E</sub>X-mode help is found by typing

```
C-h m
```

when in T<sub>E</sub>X-mode. I will list the three most important commands here. To compile (that is, `latex`) your document, use

```
C-c C-f
```

Once you have done so, you can use

```
C-c C-v
```

to view the relevant `.dvi`. Lastly, the command

```
C-c C-e
```

to `\end` an open `\begin` tag.

**2.3. A note on X.** Up until now, I've been assuming that you are able to view programs which display an X-style graphical window (e.g., `emacs`, `xdvi`). Without this ability, you must run an in-terminal editor. Alternatively, if have a Network Neighborhood shortcut to your Linux documents (mine is to `\\borg\math\user3\scoskey`), you can use any Windows programs to open your files. Also, you need to find some creative way to view your `.dvi` files. Again, with a Network Neighborhood shortcut, you could use any Windows program to access these files.

But the standard solution is to run X-Win32 (or Reflection-X, or whatever). When this program starts up, it will ask you some questions to get set up. Rather than clicking Cancel (as your instincts should dictate), fill in some random garbage until it shuts up. At this point, there will be an "X" icon in your system tray. Right-click on it and find the configuration screen. In the Keyboard tab, hit "Send left alt to X." You might find it convenient to X-Win32 to your startup folder.

Next, configure whatever secure shell client you are using (I recommend `putty`) to “Tunnel X11 Connections” or “Enable X11 Forwarding” or whatever. I’m sure you can figure it out. You may have to restart the shell.

Now, when you type `emacs` into your terminal, a separate white `emacs-y` window should appear. And bob’s your uncle.

**2.4. Figures with `xfig`.** It may seem that I’m constantly talking about `xfig`, a fantastic GNU utility for creating diagrams. I’d like to include a section right here about how to unleash the awesome power of `xfig`, but unfortunately, it’s somewhat complicated. You’ll just have to spend some time getting used to it. Check out [4] for examples of what you can do.

Another useful program for writing `.eps` figures is `dia`. I don’t know anything about it. Submissions of text to replace this section are welcome.

**2.5. Submissions with `tar`.** It is likely that your `.tex` document will require auxilliary files for `latex` to successfully compile it. For example, you might use `.eps` files for figures. In this case, you must submit *all* files required by `latex` to Jim in order for him to make heads or tails of it.

Since it is a pain to attach many files one at a time to an email, I recommend that you use `tar` to zip and compress them together. For example, to put your report and figures together you might write

```
tar czvf report.tar.gz report.tex *.eps
```

which will create a file `report.tar.gz` containing `report.tex` and anything else in the current directory ending in `.eps`. To extract it, Jim can type

```
tar xzvf report.tar.gz
```

Alternatively, if all relevant files are stored in the directory `report`, you can just `cd` to whichever directory contains it and type

```
tar czvf report.tar.gz report
```

You can use `pine` from a Linux server to turn in the file.

### 3. A NEW ENVIRONMENT

You can create your own environments for styles of text blocks that you invent. A simple environment might look like this

```
\newenvironment{rem}
{\topsep6pt% six seems to be the magic number for this
 \begin{trivlist}\item{\bf Remark}}
{\end{trivlist}}
```

and appear like this

**Remark** The earth is blue like an orange.

It takes a bit of `TEX` hacking to get something more complicated, and I am by no means an expert in this field. Here we’ve made a new theorem-like environment (it uses the theorem numbering) with a proof-box at the end of the environment, rather than the proof.

```
\newenvironment{algorithm}[1] [] % one optional arg with default value ''
{%
 \topsep6pt%
 \begin{trivlist}\item {%
```

```

    \bf Algorithm \addtocounter{theorem}{1}%
    \arabic{section}.\arabic{theorem}%
  }%
  \ifthenelse{\equal{#1}{}}{ }{ (#1)}{\bf.}%
}
{%
  \qed\end{trivlist}%
}

```

Which appears like this.

**Algorithm 3.1.** This is a new algorithm environment, which has a box at the end. □

**Algorithm 3.2** (Mine). And this is an algorithm with a name. □

**Theorem 3.3** (Yours). And a theorem with a name, for comparison.

#### 4. CUSTOMIZING COMMON COMMANDS

A number of environments have a number of optional arguments which go very under-used. Here's an example of a use of the `list` environment in its full glory. You can customize the counters and indentation very finely.

```

\newcounter{Lcount}
\begin{list}{\Roman{Lcount}}
  {\usecounter{Lcount}
  \setlength{\listparindent}{\parindent}
  }
\item ...

```

This is rendered

I This is the first item. It goes on for a while so i can see what it looks like when it extends to a second line.

*Proof.* This is not an item, but more likely a proof to the preceding item. Actually, it is the second paragraph of the preceding item.

II And this is the second item

One can also make one's own kind of funny fraction environment, as in

```

\newcommand{\myfrac}[2]{\genfrac{\left<}{\right>}{2pt}{3}{#1}{#2}}

```

This will be rendered in math mode like  $\langle \frac{th}{is} \rangle$ .

For a listing of most L<sup>A</sup>T<sub>E</sub>X environments and their options, see [7] and [5].

#### 5. A FIGURE WITH NICE FONTS

One thing you'll almost certainly want to do is draw pictures of graphs, with labelled nodes, so that people will know what the heck you're talking about in your paper. Unfortunately, this often involves the use of subscripts, and diagram editors like `xfig` typically don't provide math typesetting support.

The package `psfrag` provides a method for inserting L<sup>A</sup>T<sub>E</sub>X commands into an `xfig`-generated `.eps` file. Just place text containing something of the form

```

\tex{LaTeX stuff}

```

into your figure, and  $\LaTeX$  will perform the substitution when you compile the document. The image will not appear properly in a `dvi` viewer; you will need to compile to `postscript` to see what the figure will look like when printed.

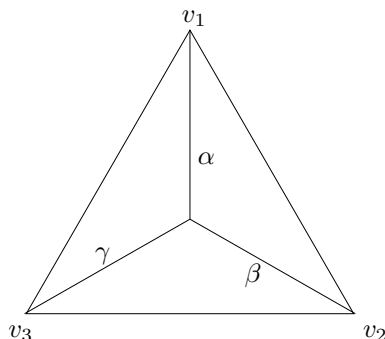


FIGURE 1. A figure with beautifully labelled boundary vertices

Note that you *must* be using a vector-graphics style figure editor such as `xfig` or Adobe Illustrator to do this! More particularly, the text to substitute must appear as actual characters in the `.eps` file.

The `psfrag` package provides several similar methods for text substitution, and many other options as well. Refer to the document [6] for complete details.

## 6. USING THE $\mathcal{AMS}$ STUFF

The American Mathematical Society provides a number of packages and add-ons for use with  $\LaTeX$ . For way too much information, see the web site [1]. For the highlights, check out the documents [2] and [3].

## 7. BIBLIOGRAPHIES WITH $\text{BIB}\TeX$

To have  $\LaTeX$  automatically sort your bibliographic entries, use `BibTeX`. To do so, first create `.bib` file, containing a formatted list of all of the references you may one day use in your life. I will not attempt to explain the format of this file; instead you can learn by reading my example `complex.bib`.

Getting the bibliography to display correctly for the first time is marginally complicated. You must have a `.bib` (say `report.bib` with some references in it, you must `\cite` one of these references in your source file, and you must have a `\bibliography` command in your source file. To get off the ground, you will have to

```
latex report
bibtex report
latex report
latex report
```

or something ridiculous like that. In the line that says `bibtex report`, note that `report` refers to the *document* name, and not to the `.bib` filename. Each time you add to or edit the `.bib` file, you will need to re-process it with

```
latex report
bibtex report
```

and go on `latex`'ing your document as usual. By default, only those references which are cited in the source file will appear in the bibliography. For more information, see the file [8].

## REFERENCES

- [1] The  $\mathcal{A}\mathcal{M}\mathcal{S}$ . *American Mathematical Society T<sub>E</sub>X Resources Home Page*. <http://www.ams.org/tex/amslatex.html>.
- [2] The  $\mathcal{A}\mathcal{M}\mathcal{S}$ . *Instructions for Preparation of Papers and Monographs*, 1999. Filename: instr-1.
- [3] The  $\mathcal{A}\mathcal{M}\mathcal{S}$ . *User's Guide for the `amsmath` Package*, 1999. Filename: amsldoc.
- [4] Timothy DeVries. Extending the medial graph to non-circular planar networks. Written for the University of Washington Mathematics REU, 2003.
- [5] Michel Goossens, Frank Mittelbach, et al. *The L<sup>A</sup>T<sub>E</sub>X Companion*. Addison Wesley, 1993.
- [6] Michael C. Grant and David Carlisle. *The PSfrag system, version 3*, 1998. Filename: pfgguide.
- [7] Sheldon Green. *Help On L<sup>A</sup>T<sub>E</sub>X Commands*. <http://www.astro.ku.dk/help/LaTeX/ltx-2.html>.
- [8] Oren Patashnik. *BibT<sub>E</sub>Xing*, 1988. Filename: btxdoc.
- [9] Keith Refson. *teT<sub>E</sub>X: A Documentation Guide*. <http://www.eng.cam.ac.uk/help/tpl/textprocessing/teTeX/newhelpindex.html>.