

# A guide to L<sup>A</sup>T<sub>E</sub>X for Maths Students

Tom Bentley

Lent 2000

## 1 What and Why?

At some point in your career as a mathematician it is almost certain that you will have to (or want to) typeset maths. The computational projects are a fine example: Although you don't have to produce your report on a computer, doing so will make for a very professional result.

Leaving aside the Why's for a moment; there are, broadly speaking, two ways to produce maths on a computer. You can use a word processor, like *Word*, or you can use a funny system called L<sup>A</sup>T<sub>E</sub>X, but pronounced 'lay-tech' so people don't think you have a fetish. In *Word* you will be able to produce your report (or whatever) in a WYSIWYG (pronounced 'wizzy-wig') environment. That is to say, you can change the appearance of the document interactively, as you go along. The problem with word processors is that they are designed to produce text. The equation editors are added as an after-thought.

In contrast L<sup>A</sup>T<sub>E</sub>X doesn't have a WYSIWYG environment. A bit like HTML, you have to specify how you want the document to look as a code which is incorporated into the content as you type it. This code is then compiled, and the document can be previewed in a previewer. It is then possible to convert the document to PostScript (.ps) and Portable Document Format (.pdf) files, or to make a hard copy. The benefit is that you will end up with a *very* smart output, in the formats that are commonly used for distributing mathematical documents.

There are a number of occasions that spring to mind, as to *why* you might want to use L<sup>A</sup>T<sub>E</sub>X as a Cambridge Maths student. These include:

- Writing your CATAM report. (*Maple* can export worksheets in L<sup>A</sup>T<sub>E</sub>X)
- Typesetting lecture notes (as has been done by the Achimedean for some courses)
- Whenever you want to publish anything containing some mathematical content.
- If you want (for some reason) to produce HTML or PDF files for use on the www. (There is a UNIX program which converts L<sup>A</sup>T<sub>E</sub>X to HTML called *latex2html*.)

In short, although using L<sup>A</sup>T<sub>E</sub>X is far from a required skill, there are times when it is very useful. The remainder of this document will detail how you can go about producing L<sup>A</sup>T<sub>E</sub>X documents.

## 2 How?

I shall assume throughout that you are using a Windows 95/98/NT PC.

It is by far the best solution to have a L<sup>A</sup>T<sub>E</sub>X system on your own computer, and the

good news is that **it's free**: Like many things on the web, you simply need to know where to look. Let us start with the ingredients:  
To get a working L<sup>A</sup>T<sub>E</sub>X system, you will need

- Your own PC
- A web connection (stricly not essential, but very helpful)
- A L<sup>A</sup>T<sub>E</sub>X compiler. and DVI previewer
- Optional: A L<sup>A</sup>T<sub>E</sub>X shell.
- GhostView
- Optional: Acrobat Reader.

### 2.0.1 A L<sup>A</sup>T<sub>E</sub>X compiler

Connect to the internet (or use an ftp client) and go to <ftp://ftp.tex.ac.uk/tex-archive/systems/win32>. You can now chose which T<sub>E</sub>X system you want to install. I personally use *miktex*, but you could opt for some thing else, like *emtex*. Anyway, you going to be downloading the latest version of (for the sake of argument) *miktex*. Unzip it. Then if you're using *miktex*, you just have to double click on the `setupwiz.exe` file. You should be confronted with a helpful little wizard for setting it up.

Which ever package you choose, you will end up with *many* directories. One of these will contain the .exe files which lie at the heart of the system. Here are some descriptions

<code>tex.exe</code>	A T <sub>E</sub> X compiler
<code>latex.exe</code>	A L <sup>A</sup> T <sub>E</sub> X compiler
<code>dvips.exe</code>	A DVI to PostScript converter.
<code>pdftex.exe</code>	A T <sub>E</sub> X/L <sup>A</sup> T <sub>E</sub> X to PDF converter
<code>pdflatex.exe</code>	A L <sup>A</sup> T <sub>E</sub> X to PDF converter (The same as pdftex.exe)

At it's very simplest, you can compile your L<sup>A</sup>T<sub>E</sub>X source code by dragging it's icon onto the latex.exe icon. Compiling L<sup>A</sup>T<sub>E</sub>X produces a DVI file. You should get a DVI viewer with our distribution of L<sup>A</sup>T<sub>E</sub>X, so you can check and see what it looks like. When you've got a finished document, so can run dvips on it to gat a PostScript version, suitable for dumping to a laser printer.

In stead of this rigmorole, you can get...

### 2.0.2 A L<sup>A</sup>T<sub>E</sub>X shell

In order to help you produce L<sup>A</sup>T<sub>E</sub>X you will probably find it useful to have a L<sup>A</sup>T<sub>E</sub>X *shell*. A shell is just a basic text editor like Windows notepad, but with some extra features built in. For example you can compile/convert your source at the click of a button and syntax-highlighting will colour different L<sup>A</sup>T<sub>E</sub>X commands a different colour so you can easily see where the commands are and where the content lies. There are a

number available, some are freeware and others are shareware.

Name	Source
4 $\TeX$	<a href="ftp://ftp.tex.ac.uk/tex-archive/systems/win32/4tex/">ftp://ftp.tex.ac.uk/tex-archive/systems/win32/4tex/</a>
$\TeX$ ed	<a href="ftp://ftp.tex.ac.uk/tex-archive/systems/win32/texed/">ftp://ftp.tex.ac.uk/tex-archive/systems/win32/texed/</a>
$\TeX$ shell	<a href="ftp://ftp.tex.ac.uk/tex-archive/systems/win32/texshell32/">ftp://ftp.tex.ac.uk/tex-archive/systems/win32/texshell32/</a>
Winedit	<a href="ftp://ftp.tex.ac.uk/tex-archive/systems/win32/winedt/">ftp://ftp.tex.ac.uk/tex-archive/systems/win32/winedt/</a>
Winshell	<a href="ftp://ftp.tex.ac.uk/tex-archive/systems/win32/winshell/">ftp://ftp.tex.ac.uk/tex-archive/systems/win32/winshell/</a>
Win $\TeX$ 2000	<a href="ftp://ftp.tex.ac.uk/tex-archive/systems/win32/wintex2000/">ftp://ftp.tex.ac.uk/tex-archive/systems/win32/wintex2000/</a>

4 $\TeX$  has a spell checker (called 4spell) which ignores  $\LaTeX$ input, you can get it from <http://4tex.ntg.nl/4tex5/4spell/>

### 2.0.3 GhostView and Acrobat Reader

You probably already have *GostView* and *GhostScript*, as they are very useful programs to have. If not, you can get them from <http://www.cs.wisc.edu/ghost/aladdin/>.

Again, you probably have *Acrobat Reader*. If not you can get it from <http://www.adobe.com/products/acrobat/readstep.html>.

## 3 And away you go....

So assuming you now have a system that works, how do you write  $\LaTeX$ ?

There are many reference guides available, so I'm not going to tell you. There are some excellent references at <http://www.eng.cam.ac.uk/help/tpl/textprocessing/>. Those should be more than adequate for your needs:  *$\LaTeX$  is not hard*.

One of the best ways of learning is to cannibalise other people's code. To this end there is a file called short.tex in this directory. If you [download it](#) you can try and compile it (to get [\[DVI\]](#), [\[PS\]](#) or [\[PDF\]](#)), or you can just [look at it](#).