

An Introduction to \LaTeX

Iryna Schlackow
Mathematical Institute

This talk and other useful \LaTeX -related information is available at
<http://www.maths.ox.ac.uk/help/faqs/latex/>

May 21, 2009

The Name of the Game

\TeX is a computer program created by Donald E. Knuth. It is aimed at typesetting text and mathematical formulae. Knuth started working on it in 1977, and \TeX as we use it today was released in 1982. \TeX is renowned for being extremely stable and virtually bug free. The version number of \TeX is converging to π and is now at 3.1415926.

\TeX is pronounced “Tech,” with a “ch” as in the German “Ach.” In an ASCII environment, \TeX becomes TeX.

The Name of the Game

\LaTeX is a \TeX macro package which enables authors to typeset and print their work at the highest typographical quality, using a predefined, professional layout. \LaTeX was originally written by Leslie Lamport in 1980s, and its current version, $\LaTeX 2_{\epsilon}$, was released in 1994.

\LaTeX (LaTeX in an ASCII environment) is pronounced “Lay-tech” or “Lah-tech.” $\LaTeX 2_{\epsilon}$ (LaTeX2e) is pronounced “Lay-tech two e”.

Why L^AT_EX?

- The typesetting of mathematical formulae is supported in a convenient way.
- Complex structures (footnotes, cross-references, bibliography) can be generated easily.
- Professionally crafted predefined layouts are available so that users only need to learn a few commands that specify the logical structure of a document.
- It is easy to superimpose another document style, designed by a publisher, on a L^AT_EX file.
- Many scientific journals accept manuscripts in L^AT_EX only.
- T_EX is free and highly portable. Therefore the system runs on almost any hardware and software platform available.

Why \LaTeX ?

- \LaTeX is built on a programming language and is therefore extensible. Free add-on packages exist for many typographical tasks not directly supported by basic \LaTeX :
 - customised class styles for scientific journals,
 - presentation class styles,
 - packages for writing CVs and cover letters,
 - packages for music typesetting,
 - packages for writing linguistic papers,
 - ...

Any disadvantages?

- Not (traditionally) a WYSIWYG system.

\LaTeX vs WYSIWYG Systems

WYSIWYG systems (Word)

The output is precisely what you type in.

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

The document layout is specified by means of *visual design*.

You may spend too much time fiddling with fonts and margins. The document is likely to have little or inconsistent structure.

\LaTeX (traditional approach)

You type in \LaTeX “code” which needs to be compiled to produce the final output.

```
\left(\begin{array}{ll}
1 & 2 \\
3 & 4
\end{array}\right)
```

A suitable layout is chosen by \LaTeX once the *logical structure* of the document has been specified.

It is very hard to write unstructured and disorganised documents in \LaTeX .

L^AT_EX vs WYSIWYG Systems

Scientific Word is a L^AT_EX-based WYSIWYG system for Windows.
LyX is a WYSIWYM system for Linux, Unix, Mac OS X and MS Windows.

Input File

Every L^AT_EX input file possesses a certain structure. You start with specifying what sort of document you intend to write

```
\documentclass[options]{class}
```

This is followed by the preamble where you can include commands influencing the style of the whole document or load packages adding new features to the L^AT_EX system

When all the setup work is done you start the body of the text.

```
\begin{document}
```

Now you enter the text mixed with L^AT_EX commands. At the end of the document you add

```
\end{document}
```

Anything that follows this command will be ignored by L^AT_EX.

Output File

Once the L^AT_EX file is compiled, an output file is produced. It can be a `.dvi` or a `.pdf` file:

- if the file is compiled with the `latex` command, it usually produces a `.dvi` file,
- if the file is compiled with the `pdflatex` command, it usually produces a `.pdf` file.

Many L^AT_EX editors can produce both `.dvi` and `.pdf` files.

L^AT_EX Editors

L^AT_EX Editor (LEd) is a very good free editor for Windows.

- built-in DVI viewer which provides two-way navigation between the source text and the preview page,
- built-in spellchecker,
- descriptive hints,
- command completion,
- easy navigation between labels and references, bibliography items and citations,
- related opening and closing parentheses are highlighted with the same colour,
- user-friendly interface.

L^AT_EX Editors

WinEdt is a commercial editor for Windows.

Other L^AT_EX editors include TeXnikCenter (for Windows), iTeXMac2 (for Mac OS X), Kile, TeXmaker (runs on Unix, Mac OS X), Emacs (runs on most systems).

The `\documentclass` command

```
\documentclass[options]{class}
```

Document classes: article, book, letter, beamer, ...

Options (need to be separated by commas): 10pt, 11pt, 12pt, a4paper, twoside, landscape, fleqn, ...

For example, the command

```
\documentclass[11pt, twoside, a4paper]{article}
```

instructs L^AT_EX to typeset the document as an article with a base font size of 11pt, and to produce a layout suitable for double side printing on A4 paper.

Preamble

The preamble tells L^AT_EX which packages to use.

```
\usepackage[options]{package}
```

Examples of packages include `amsmath`, `amsthm`, `color`.

The preamble also defines the style of the document. You can change margins, width or height of text, indentation, ...

```
\setlength{\textheight}{600pt}
```

```
\addtolength{\textwidth}{-1cm}
```

```
\setlength{\parindent}{0pt}
```

Special Characters

The following symbols are reserved characters that have a special meaning in L^AT_EX and, when entered directly in text, will coerce L^AT_EX to do things you did not intend

$\$ \& \% \# _ \{ \} \sim \hat \ \backslash$

You can produce these characters in the text.

For $\$$, $\&$, $\%$, $\#$, $_$, $\{$ and $\}$ type a backslash \backslash in front of them.

For \sim , \hat and \backslash you need to use special commands.

L^AT_EX Commands

L^AT_EX commands are case-sensitive and consist of a backslash `\` followed by

- a string of letters, or
- exactly one non-letter (e.g. a special character).

Commands may have

- no arguments,
- mandatory arguments, which are input in braces `{ }`,
- optional arguments, which are input in square brackets `[]`.

For example,

```
\framebox[3in]{In \LaTeX, you can write in \textit{italic}.}
```

produces

In L^AT_EX, you can write in *italic*.

Macros

You can define your own commands using the following command:

```
\newcommand{name}[num]{definition}
```

For example, in order to produce $\frac{\partial L}{\partial x}$ we need to type

```
\frac{\partial L}{\partial x}
```

whereas if we put the following line in the preamble

```
\newcommand{\pd}[2]{\frac{\partial #1}{\partial #2}}
```

it is enough to type

```
\pd{L}{x}
```

If the command name is already defined, use `\renewcommand`.

E.g. `\renewcommand{\leq}{\leqslant}` replaces \leq with \leqslant .

Sections

The following sectioning commands are available for the article class: `\section`, `\subsection`, `\subsubsection`, `\paragraph`, `\subparagraph`.

```
\section{title}
```

or, if the title is too long,

```
\section[Title for the table of contents]{A longer  
title, shown in the text}
```

To create a table of contents, use the `\tableofcontents` command.

To create an unnumbered section that does not appear in the table of contents, use the starred versions of the commands.

```
\section*{title}
```

Environments

Environments are building blocks of a L^AT_EX file. Each declaration of environment has the following syntax

```
\begin{environment} text \end{environment}
```

Examples include

- `\begin{center} ... \end{center}`,
- `\begin{document} ... \end{document}`,
- `\begin{enumerate} ... \end{enumerate}`.

Environments can be nested within each other as long as the correct nesting order is maintained:

```
\begin{aaa}  
  ...  
  \begin{bbb}  
    ...  
    \end{bbb}  
  ...  
\end{aaa}
```

Lists

To produce a bullet point list, use the `itemize` environment. To produce a numbered list, use the `enumerate` environment. Each item of a list begins with the `\item` command.

```
\begin{enumerate}  
  \item[label (optional)]  
  \item[label (optional)]  
\end{enumerate}
```

To change the default style of numbering, redefine the commands `\labelenumi`, `\labelenumii`, `\labelenumiii`, `\labelenumiv` that provide styles for the four allowed levels of nesting. E.g.

```
\renewcommand{\labelenumi}{\Alph{enumi}}  
\renewcommand{\labelenumii}{(\roman{enumii})}
```

put in the preamble (inside an enumeration environment) produce lists numbered A, B, C ... for the first level and (i), (ii), (iii) ... for the second level. The change will be applied to the whole document (to this specific enumeration environment).

Typesetting Mathematics

Mathematical text within a paragraph is entered between `$` and `$`, e.g., `$\alpha=\sum_{i=1}^n \beta^i$` produces $\alpha = \sum_{i=1}^n \beta^i$.

To insert an equation on a separate line, write `\[... \]` or `\begin{displaymath} ... \end{displaymath}`, e.g.

```
\[ \alpha = \sum_{i=1}^n \beta^i \]
```

produces

$$\alpha = \sum_{i=1}^n \beta^i.$$

To produce in line formulae with `\displaymath` layout, use `$\displaystyle...$` instead of `$...$`, e.g. from the example

above we get $\alpha = \sum_{i=1}^n \beta^i$.

To number your equation, use the `equation` environment instead of the `displaymath` one.

Vertically Aligning Equations

For formulae running over several lines or simultaneous equations use the `eqnarray` environment, e.g.

```
\begin{eqnarray}
  \alpha & = & 7+x \\
  \beta+\gamma & = & \int_{0}^{\infty} p(x) \, dx
\end{eqnarray}
```

produces

$$\alpha = 7 + x \tag{1}$$

$$\beta + \gamma = \int_0^{\infty} p(x) dx. \tag{2}$$

You can align the equations about any symbol.

To produce unnumbered equations, use the `eqnarray*` environment. To not number only certain equations in an array, insert the command `\nonumber`.

Cross-references

L^AT_EX provides the following commands for cross-referencing:
`\label{marker}`, `\ref{marker}`, `\eqref{marker}` and
`\pageref{marker}`.

For example,

```
\begin{thm} \label{thm:one_plus_one}
  \begin{equation} \label{eqn:one_plus_one}
    1 + 1 = 2.
  \end{equation}
\end{thm}
```

Let us now prove equation `\eqref{eqn:one_plus_one}`
from Theorem `\ref{thm:one_plus_one}`.

produces

Theorem 1.

$$1 + 1 = 2. \tag{1}$$

Let us now prove equation (1) from Theorem 1.

Other L^AT_EX features

In fact, all this was just the tip of the iceberg. With L^AT_EX you can do *much* more:

- easily generate complex structures such as footnotes, fancy headers, title pages or bibliographies;
- produce vertically aligned material, e.g. tables or matrices;
- produce diagrams;
- manipulate counters;
- insert graphics (and even movies);
- write text in various languages;
- insert hyperlinks into the output pdf documents;
- change layout of the whole document as well as individual sections/pages/lines,
- ...

- Let \LaTeX do as much of your work as possible. Make use of
 - labels and cross-references,
 - enumeration environments,
 - sectioning commands,
 - array environments,
 - macros.

- Distinguish between *italic* and roman fonts in math mode. Compare

$$\int_0^1 e^{inx} \cos nx dx = 1 \quad \text{for } n = 0$$

and

$$\int_0^1 e^{inx} \cos nx dx = 1 \quad \text{for } n = 0.$$

Use roman alphabet for

- non-mathematical symbols,
- differential d , exponential e and complex i ,
- functions like \sin , \cos , \log etc. (for these you need to use the backslash version, i.e. write `\cos` instead of `cos`).

- Add punctuation after equations and inside enumeration environments.
- Differentiate between
 - hyphen X-ray,
 - en-dash pages 1–12, Cauchy–Schwartz inequality, and
 - em-dash a punctuation dash — like this.

How many authors does the Birch–Swinnerton-Dyer conjecture have?

- Do not use " for quotation marks. Instead type
 - two ` (grave accent) for opening quotation marks, and
 - two ' (vertical quote) for closing quotation marks.
- Don't be frustrated if something doesn't work out — Google is always there to help you!