

Introduction to L^AT_EX

or

Why Word suck

Simen Hagen

simen.hagen@iu.hio.no

March 15, 2005

WYSIWYG

What You See Is What You Get

Include programs like Word, OpenOffice, FrameMaker etc.

Pros

- Easy to use for beginners
- You get what you want (usually)
- Usable for simple letters, memos, flyers etc.

Cons

- You have to be the designer
- Hard to be consistent

Structure Oriented Writing

Include “programs” like L^AT_EX, scribe, html, xml.

Pros

- Concentrate on the subject
- Easy to get a consistent, good design
- Best suited for articles, reports, books, mathematical formulas etc.

Cons

- Many commands to learn
- The commands are in the text
- Awkward in the beginning

TEX

- was created by *Donald Knuth*
- is freely available
- is extendable

L^ATEX

- was created by *Leslie Lamport*, and is based on TEX
- adds macros, packages, indexes etc.
- is more user friendly than TEX
- is easily extendable

The Preamble – An Example

```
\documentclass[11pt , a4paper , english]{ article }
\usepackage[ latin 1]{ inputenc }
\usepackage[T1]{ fontenc }
\usepackage{ babel }
\usepackage{ url }

\title{Literature Survey}
\author{Simen Hagen\\ Oslo Univeristy College\\
  \url{simen.hagen@iu.hio.no}}
\date{8th March, 2005} %% Leave this out and it will use todays date.
```

The body – An Example

```
\begin{document}  
\maketitle
```

```
\begin{abstract}  
  In this section I write my abstract. An abstract is the authors  
  version of a short summary. It often includes a short “guide” to  
  the article.  
\end{abstract}
```

```
\section{Introduction}
```

This is a literature survey on a topic that I have chosen.

```
\section{A chapter}
```

In this part I will write some important information about the topic that I have chosen.

```
\section{Conclusion}
```

Now it is time to wrap up and think of something very impressive to say.

```
\end{document}
```

The Resulting Page

Literature Survey

Simen Hagen
Oslo Univeristy College
`simen.hagen@iu.hio.no`

8th March, 2005

Abstract

In this section I write my abstract. An abstarct is the authors version of a short summary. It often includes a short "guide" to the article.

1 Introduction

This is a literature survey on a topic that I have chosen.

2 A chapter

In this part I will write some important information about the topic that I have chosen.

3 Conclusion

Now it is time to wrap up and think of something very impressive to say.

How to run L^AT_EX

You can use any text editor to write your L^AT_EX files. The L^AT_EX file typically ends with the extension *.tex*. I would recommend *Emacs* since it has many functions to help you write your “code”.

When you have written the document, you run the L^AT_EX program with the following command:

```
$ latex <my_latex_file>.tex
```

If all goes well, your code compiles without errors, and it produces a *.dvi* document. You can view this document with any *dvi* viewer, e.g. *xdvi* in a UNIX environment, or with *yap* if you are using windows (and the MiKTeX-package for windows).

You can also convert your *.dvi* files into *.ps* with *dvips*, or to *.pdf* with *dvipdf*.

You could also use the command *pdflatex* instead of the “normal” *latex* to produce a *.pdf* directly.

Document Classes

Any document written in L^AT_EX need to have a document class. The document class defines how the document will be rendered on the page. The example on the previous slides used the *article* style, but there are others available. The most common ones are:

- article
- report
- book

There is also a *letter* class, and more can be downloaded or you can create your own.

For your survey, the *article* class should be used.

Classes and properties

There are a number of properties you can set that influences the document created. The syntax for the command is

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

The class is as on the previous slide, and you can have several options, dependent on the class you have chosen. Below is a list of a few of the possibilities.

10pt	11pt	12pt	letterpaper
a4paper	onecolumn	twocolumn	oneside
twoside	draft	final	

Writing Text

- You write text just as normal, plain text.
- There are a few reserved characters (see table below).
- Commands start with a `\`.
- Commands can occur anywhere in the text.
- You signal a new paragraph with a blank line.

Character	Write	Character	Write
#	<code>\#</code>	}	<code>\}</code>
\$	<code>\\$</code>	-	<code>\-</code>
%	<code>\%</code>	^	<code>\textasciicircum</code>
&	<code>\&</code>	~	<code>\textasciitilde</code>
{	<code>\{</code>	\	<code>\textbackslash</code>

Sections

It is common practice to divide a document into parts. In \LaTeX there are several commands, taking the form $\backslash section_command\{title\}$. These commands are:

part	chapter	section	paragraph
		subsection	subparagraph
		subsubsection	

These commands produce sectioning numbers, and are automatically added to a table of context.

There is also a version of the sectioning commands, on the form $\backslash section_command*\{title\}$, that do not produce sectioning numbers, or produce an entry into the table of context.

Text Formatting

You can mark text in several different ways. Here is a list of the most common ways:

Command	Result
<code>\emph{...}</code>	<i>Italic</i>
<code>\textbf{...}</code>	Bold
<code>\textsc{...}</code>	CAPITALISE
<code>\texttt{...}</code>	Typewriter

It is also possible to combine the text formatting to have ***BOLD CAPITALISE*** or ***Italic Bold***.

Creating Lists

```
\begin{itemize}  
  \item Item number one  
  \item Item number two  
  \item Item number three  
  \item And so on...  
\end{itemize}
```

- Item number one
- Item number two
- Item number three
- And so on...

```
\begin{enumerate}  
  \item Item number one  
  \item Item number two  
  \item Item number three  
  \item And so on...  
\end{enumerate}
```

1. Item number one
2. Item number two
3. Item number three
4. And so on...

Creating Lists... Continued

```
\begin{description}  
  \item[Term 1] Term number one  
  \item[Term 2] Term number two  
  \item[Term 3] Term number three  
  \item[Term 4] And so on...  
\end{description}
```

Term 1 Term number one
Term 2 Term number two
Term 3 Term number three
Term 4 And so on...

Verbatim Text

Sometimes it is necessary to type something that should appear exactly as typed, without \LaTeX formatting it. You can achieve this by using verbatim text. The text will appear exactly as typed, in a typewriter font.

Using `\verb+my text+` gives you verbatim text within a line. It will quote all text that appears within the '+'-markers. You can use any marker you like, not just '+', as long as the marker do not occur in the text you are quoting.

You can also create an environment, like this:

```
\begin{verbatim}  
  \textbf{Some tekst.}  
\end{verbatim}
```

```
\textbf{Some tekst.}
```

Program Listings

Sometimes you want to refer to a program file or some code that you have written. You can use `\verb` for this, but you can also use the package `listings`. It will pretty print your code for you, and it support most popular languages, including `LATEX`.

```
\lstset{language=[LaTeX]TeX}
\lstset{basicstyle=\footnotesize}
\lstinputlisting[firstline=1, lastline=12]{example1.tex}

\begin{lstlisting}
... some code ...
\end{lstlisting}
```

Math in L^AT_EX

It is easy to write formulas in L^AT_EX. You can write inline formulas like $A = \sqrt{B^2 + C^2}$. This is written like `$A = \sqrt{B^2 + C^2}$`.

You can also write “bigger” formulas that is written on a separate line. Then you would write:

```
\begin{equation}
  A = \sqrt{B^2 + C^2}
\end{equation}
```

to get the the following:

$$A = \sqrt{B^2 + C^2} \quad (1)$$

As you can see, this equation have a number, and this number can be referenced.

```
\begin{displaymath}
  A = \sqrt{B^2 + C^2}
\end{displaymath}
```

Which gives you this:

$$A = \sqrt{B^2 + C^2}$$

In this case there is no equation number. You could also use the short hand version that can be written `$$... $$`.

Using Illustrations

It is easy to include graphics in \LaTeX , but you have to include an extra package to do it.

```
\usepackage{graphicx}
```

Then, somewhere in your document you write:

```
\includegraphics[height=3cm]{some_picture}.
```

\LaTeX , unfortunately, only understands .eps pictures (Encapsulated PostScript). If you use pdf \LaTeX , it understands a few more formats (.jpg, .png and .pdf). You do not have to specify the extension.



Floating Figures

You can create floating figures. \LaTeX will place a floating figure wherever it finds room for it. It is the preferred way of doing tables, pictures etc.

Using the example from the previous slide, we can create a floating figure like this:

```
\begin{figure}[t]
  \centering
  \includegraphics[height=3cm]{some_picture}
  \caption{This is my little picture}
\end{figure}
```

If you make your images and figures floating, you will save yourself a lot of problems later on.

Cross references

Often, we want to refer to something in our text, like a figure or table. And often, particularly if we are using floating figures, we do not know where they are, exactly. Enter the cross reference.

Before we can reference anything, we have to create a label. If we expand the example from the previous slide, we can create a label like this:

```
\begin{figure}[t]
  \centering
  \includegraphics[height=3cm]{some_picture}
  \caption{This is my little picture}\label{picture}
\end{figure}
```

We can now reference the page that the label occur on by using `\pageref{picture}`, or the figure number by using `\ref{picture}`.

For more diverse reference-naming, see the package `varioref`.

BIB_TE_X

BIB_TE_X is an external program that helps you manage your references when writing in L_AT_EX.

First you have to create a database of your references. You can then use L_AT_EX commands to access the reference through a *key* that you have defined.

When you have written your document and made your references, you start by running L_AT_EX on your document. You will get a couple of messages complaining that there are missing references, but that is normal. Now you run BIB_TE_X on that file, and finally you run L_AT_EX **two** more times. You should now have your bibliography.

Creating the database

At some point you have to create the database. It is just a text file (with a .bib extension), formatted in a particular way. There are several different types of entries in the database (see for example <http://www.cs.arizona.edu/~collberg/Teaching/07.231/BibTeX/bibtex.html> for a list).

Some of the more common are:

article An article from a journal or magazine.

book A book with an explicit publisher.

proceedings An article in a conference proceedings.

techreport A report published by a school or other institution, usually numbered within a series.

The Entries

In the database, each entry starts with an '@', and all the fields in the database is enclosed within curly-braces ('{' and '}'). Below is a sample entry:

```
@techreport{Mega,  
  author =      {Siri Fagernes and Simen Hagen},  
  title =      {The {MEGA} Project – Virtual Emotions in Mixed  
                Reality},  
  institution = {Telenor Research and Development},  
  year =      2002,  
  number =     {N 50/2002},  
  address =    {Fornebu , Norway}  
}
```

Referring

When we have created an entry and want to refer to it, we use the `\cite[extra]{key}` command.

This first reference is created with the command `\cite{MEGA}: [1]`. The second reference is created with `\cite[page~2]{MEGA}: [1, page 2]`.

References

- [1] Siri Fagernes and Simen Hagen. The MEGA project - virtual emotions in mixed reality. Technical Report N 50/2002, Telenor Research and Development, Fornebu, Norway, 2002.

Reference style

There are four basic styles available for how your referencing and bibliography will look. They are *plain*, *unsrt*, *abbrv* and *alpha*. There is also a host of other styles that you can install and use in your documents, but these four should be present on all systems.

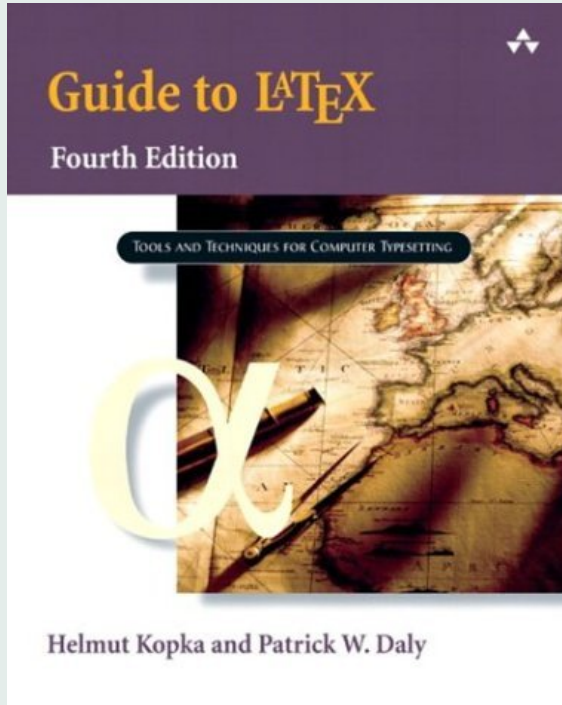
You can set which style by using the command `\bibliographystyle{<style>}`.

To insert the bibliography, you give the command `\bibliography{<bib-databases>}`, where `<bib-databases>` is a comma-separated list of the database files that you use in your document.

So, given that your database is called *example*, you could add the following commands where you want the bibliography to appear.

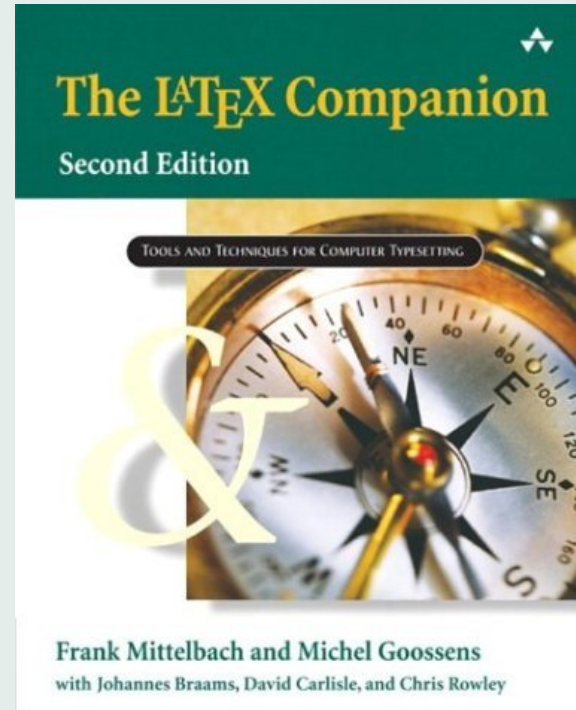
```
\bibliographystyle{plain}  
\bibliography{example}
```

Books



Ok

This is a nice introduction and reference book to L^AT_EX. It has lots of examples and useful information.



This is a very good reference book, and is quite a bit more advanced than the “Guide to L^AT_EX”. It is considered to be THE reference to L^AT_EX.

More Information

- Dag Langmyr has some good \LaTeX references in Norwegian at his homepage <http://heim.ifi.uio.no/~dag/>.
- You can find a simple introduction to \LaTeX at <http://www.cs.hmc.edu/qref/latex/qref/>.
- There is a good hypertext reference at <http://www.giss.nasa.gov/latex/ltx-2.html>.
- \LaTeX by Example is another easy to use reference: <http://cs5.ecok.edu/ByExample/ByExample.html>.

Even More Information

- There is a L^AT_EX FAQ at <http://www.tex.ac.uk/cgi-bin/texfaq2html>.
- AUCT_EX is a package for GNU Emacs and XEmacs for writing T_EX and L^AT_EX files: <http://www.gnu.org/software/auctex/>.
- CTAN is the “Comprehensive T_EX Archive Network”, and has a lot of information about T_EX and L^AT_EX: <http://www.ctan.org/>.
- CTAN also has “The Not So Short Introduction to LaTeX”, also known as “LaTeX in 131 minutes”. <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>.

You will also find these links at the course-homepage.