



L^AT_EX

ES 2016/2017

Adam Belloum

Material Prepared by Eelco Schatborn

A.S.Z.Belloum@uva.nl

September 1, 2016



ESA: L^AT_EX

Monday:

1. Intro & Hello World
2. Document class, options, packages
3. Sections, lists, and some more
4. Documentation



ESA: L^AT_EX

Today:

1. Maths
2. Tables and figures
3. Cross-references, citations
4. Defining commands and environments

Recap: \LaTeX philosophy

Working on content, not layout means having to automate typesetting. This is where \LaTeX comes in and extends \TeX .

Should you have to “tweak” the layout just remember:

- ▶ don't use ‘quick fixes’
- ▶ use your own command (re)definitions in the preamble
- ▶ use comments in your source, especially if it is large

Special symbols: Accents

Examples:

`\'e, \'a, \'e, \'o` gives é, á, è, ò
`\"o, \"e` gives ö, ë

you can use `\usepackage[utf8]{inputenc}`

à l'Université de Compiègne

You can also create very strange symbols in this way:

`\'w, \'r, \'s, \'t` gives ẃ, ŕ, ś, t́
`\"q, \"r` gives q̈, r̈

— The `babel` package offers extra support for various languages



Math mode

In math mode \LaTeX behaves differently than in paragraph mode; spaces are **ignored**, there are (a lot of) other commands available, ...

We'll discuss three environments for math mode.

- ▶ `math`
- ▶ `displaymath`
- ▶ `equation`

All environments can accept the same math commands, they just show them differently.

The math environment

To show math symbols and formula's in your document you can use the math environment: `\begin{math}...\end{math}`

- ▶ `$` and `\(` are short forms for `\begin{math}`
- ▶ `$` and `\)` are short forms for `\end{math}`

Example

for all k ; $(k^2 \geq 1)$ if $k \neq 0$.

for all k ; $k^2 \geq 1$ if $k \neq 0$.

The `displaymath` environment

To show unnumbered, separate formulas use the `displaymath` environment: `\begin{displaymath}... \end{displaymath}`

- ▶ `\[` and `\]` are short forms for `\begin{displaymath}` and `\end{displaymath}` respectively

Example

for all k ;
if $k \neq 0$.

for all k ;

$$k^2 \geq 1,$$

if $k \neq 0$.

The equation environment

To show numbered, separate formulas use the equation environment: `\begin{equation}...` `\end{equation}`

Example

Voor alle k geldt dat

```
\begin{equation}
```

```
  k^2 \geq 1,
```

```
\end{equation}
```

mits $k \neq 0$.

Voor alle k geldt dat

$$k^2 \geq 1, \tag{1}$$

mits $k \neq 0$.

Math: remarks (1)

- ▶ spaces are **ignored**: `$1 2 3 x y z$` gives $123xyz$
- ▶ sub- and superscripts:

$$x^2 = y^{a \times b} \quad x^2 = y^{a \times b}$$

$$x^i_n = y_{k,l,m} \quad x^i_n = y_{k,l,m}$$

$$x^{y^2} \quad x^{y^2}$$

- ▶ ligatures: `$different$` versus `$\mathit{different}$`
different versus *different*



Math: remarks (2)

- ▶ `\mathbf{}`, `\mathrm{}`, `\mathit{}`
- ▶ ranges: `x_1, x_2, \ldots, x_n` gives x_1, x_2, \dots, x_n
- ▶ the greek alphabet:
`$\alpha, \beta, \ldots, \omega$` gives $\alpha, \beta, \dots, \omega$
`Γ, \ldots, Ω` gives Γ, \dots, Ω



Math: remarks (3)

To create fractures use `\frac{}{}:`

`\[x = \frac{x + z/2}{y^2 + 1} \]`

gives



Math: remarks (3)

To create fractures use `\frac{ }{ }`:

`\[x = \frac{x + z/2}{y^2 + 1} \]`

gives

$$x = \frac{x + z/2}{y^2 + 1}$$

Or use `$x = \frac{x + z/2}{y^2 + 1}$` in your text.

Or use $x = \frac{x+z/2}{y^2+1}$ in your text.

Lengths and sizes

\LaTeX provides the following units to use in commands:

`mm` millimetre $\approx 1/25$ inch

`cm` centimetre = 10 mm

`in` inch = 25.4 mm

`pt` point $\approx 1/72$ inch ≈ 1 mm

`em` approx width of an 'M' in the current font

`ex` approx height of an 'x' in the current font

- ▶ `\hspace{}` and `\vspace{}` can be used to change the layout of a page.
- ▶ use `\smallskip`, `\medskip` or `\bigskip` for predefined vertical spacing
- ▶ you can create new and change existing length variables (like commands)



Floating bodies...

- ▶ **figures** and **tables** need special attention since they **cannot be broken across pages**
- ▶ they generate “floating bodies” (bit like floating boxes)
- ▶ L^AT_EX will **find** a place for them on the page for **optimum legibility**
- ▶ they will be placed as **close** as possible to their definition but **no guarentees**
- ▶ most commands that generate floating bodies accept positional *preferences*



The figure environment (1)

```
\begin{figure}[pos]
% the figure
\caption{A picture of a cow} %% optional
\label{fig:cow} %% optional
\end{figure}
```

- ▶ using pos you can provide a *preference* for placement: **h** (here), **t** (top), **b** (bottom), **p** (page), or a **combination**
- ▶ the figure environment can contain just about anything
- ▶ you can also give a caption above the content of the figure.

Note a label *must* be specified *after* the caption



The picture environment (1)

- ▶ the `picture` environment can be used to draw simple pictures
- ▶ 'native' to \LaTeX , no external packages are needed
- ▶ you can draw circles, lines, arrows, etcetera together with text

Downsides

- ▶ works only with **explicit co-ordinates**, there are restrictions to the gradients of lines
- ▶ takes a lot of preparation, you should work on paper first. . .

The picture environment (2)

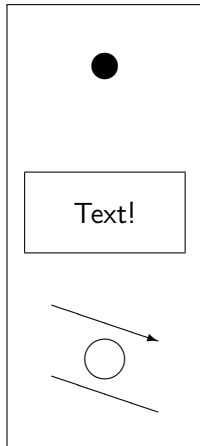
Example

```
\setlength{\unitlength}{1pt}  
\fbox{  
  \begin{picture}(60,160)(0,0)  
    \put(0,70){\framebox(60,30){Text!}}  
    \put(30,30){\circle{15}}  
    \put(30,140){\circle*{10}}  
    \put(50,10){\line(-3,1){40}}  
    \put(10,50){\vector(3, -1){40}}  
  \end{picture}  
}
```

The picture environment (2)

Example

```
\setlength{\unitlength}{1pt}
\fbbox{
\begin{picture}(60,160)(0,0)
\put(0,70){\framebox(60,30){Text!}}
\put(30,30){\circle{15}}
\put(30,140){\circle*{10}}
\put(50,10){\line(-3,1){40}}
\put(10,50){\vector(3,-1){40}}
\end{picture}
}
```



The tabular environment (1)

You can create **tables** using the `tabular` environment:

```
\begin{tabular}[pos]{table spec}... \end{tabular}
```

- ▶ cell contents can be left (`l`) or right (`r`) aligned, centered (`c`) or justified (`p{}`)
- ▶ cells can be divided by lines using `|` and `\hline`
- ▶ inter-column spacing can be replaced using `@{}`, this comes in handy for aligning numeric columns
- ▶ use `\multicolumn{num}{spec}{title}` to create a multicolumn column



The tabular environment (2)

Example

Pi expr.	Value
π	3.1416
π^π	36.46
$(\pi^\pi)^\pi$	80662.7

Example taken from NSSIT \LaTeX .



The tabular environment (2)

Example

```
\begin{tabular}{c|r @{.} l}  
Pi expr.      &  
\multicolumn{2}{c}{Value} \\  
\hline  
$\pi$          & 3.1416 \\  
$\pi^{\pi}$    & 36.46 \\  
$(\pi^{\pi})^{\pi}$ & 80662.7 \\  
\end{tabular}
```

Pi expr.	Value
π	3.1416
π^{π}	36.46
$(\pi^{\pi})^{\pi}$	80662.7

Example taken from NSSIT \LaTeX .

Creating new commands

To add your own commands, use the
`\newcommand{name}[num]{definition}` command.

Example

`\newcommand{\shout}[1]{\{\large\textbf{\shout{#1}}\}}`

can be used as follows:

Let's `\shout{shout}` it out!

which produces:

Let's **shout** it out!

Creating new environments

To add your own environment, use the
`\newenvironment{name}[num]{before}{after}` command.

Example

```
\newenvironment{king}  
  {\rule{1ex}{1ex}}  
  {\rule{1ex}{1ex}}
```

can be used as follows:

```
\begin{king}  
  I rule!           produce:  
\end{king}
```


Creating new environments

To add your own environment, use the
`\newenvironment{name}[num]{before}{after}` command.

Example

```
\newenvironment{king}  
  {\rule{1ex}{1ex}}  
  {\rule{1ex}{1ex}}
```

can be used as follows:

```
\begin{king}  
  I rule!           producing:  ■ I rule!  ■  
\end{king}
```



Errors in L^AT_EX

- ▶ error given by latex are put into a .log file
- ▶ syntax errors imply the end of a compilation run
- ▶ you'll get a lot of **overfull** and **underfull hboxes** errors
- ▶ the **draft** option to the standard classes will make these **visible** in the output



Cross-references

By adding a `\label{}` at certain points you can refer to that point later on: `\section{Introduction}\label{sec:intro}`

Use `\ref` to refer to that section by its number:

In Section `\ref{sec:intro}` we'll start off\ldots
becomes

In Section we'll start off...



The auxiliary file

- ▶ every time you run latex a `verb.aux` file is created or refreshed
- ▶ it contains among others the names of **sections**, **information** on **labeled** objects, **bibliography** information, etcetera.
- ▶ you'll have to **run** latex **twice** to propagate changes in the auxiliary file to your output correctly.
 - ▶ the **first run** is to **create** or **change** the auxiliary file
 - ▶ the **second run** serves to **propagate** those changes into the output



Table of contents

You can generate a table of contents using the command

`\tableofcontents`

The table will be created in its place. The TOC information is taken from the **auxiliary file** and placed into a `.toc` file. Note that you will have to **run** latex **twice** to generate the information correctly.

The bibliography

You can create a bibliography (list of references) inside the `thebibliography` environment, using a `\bibitem` command for each entry:

```
\begin{thebibliography}{99}
```

```
\bibitem{companion}
```

M. Goossens, F. Mittelbach, et al.

`\emph{The \LaTeX\ Companion}`.

Addison-Wesley. 1994.

```
\bibitem{texbook}
```

D. Knuth.

Remarks to Celebrate the Publication of Computers & Typesetting.

```
\end{thebibliography}
```



The result

Bibliography



M. Goossens, F. Mittelbach, et al. *The L^AT_EX Companion*. Addison-Wesley. 1994.



D. Knuth. Remarks to Celebrate the Publication of Computers & Typesetting. *TUGboat*, 7(2):95–98. 1986.

—

Use `\cite` to refer to the bibliography using a key:

In `\cite{texbook}` the author writes\ldots gives:
author writes. . .

In [2] the

Bibliography using BibT_EX

- ▶ a more advanced way of maintaining and generating a bibliography is by using the BibT_EX package
- ▶ included in most distributions
- ▶ BibT_EX also uses the **auxiliary** file but uses a **.bib** file for its entries
- ▶ it will **only include** those **entries** that are used in the document
- ▶ you can use the same **.bib** library for years to come
- ▶ different styles are available to be used for the layout of the list
- ▶ still uses **\cite** for its citations



Splitting the source (1)

- ▶ when working on **large documents** **splitting** the source into smaller pieces can make writing easier
- ▶ you can split into **chapters** or **sections** which can be put into **different files**
- ▶ the separate files can be included into a main source file using the `\input{}` command
- ▶ the `\input{}` command merges the content of its argument file with the source

Splitting the source (2)

Example

```
\documentclass{article}  
\input{myPreamble}  
\begin{document}  
  \input{deel1}  
  \input{deel2}  
  \input{deel3}  
\end{document}
```



Splitting the source (3)

The `include` command

- ▶ you can also **import** source files using the more advanced `\include{}` command
- ▶ you can use the `\includeonly{}` command to specify which files to include
- ▶ **auxiliary** information will always be available, even if you exclude one of the files temporarily

Splitting the source (4)

Example

```
\includeonly{chap1,chap3}  
\documentclass[a4paper,12pt]{report}  
\begin{document}  
\include{titlepage}  
\include{chap1}  
\include{chap2}  
\include{chap3}  
\end{document}
```

The graphicx package

- ▶ First add `\usepackage{graphicx}` to your preamble.
- ▶ There are several options for drivers etcetera.
- ▶ To include pictures into your document you can use the `\includegraphics[width]{filename}` command.



Example

```
\includegraphics[width=3cm]{ibm-mainframe.jpg}
```

The command can work with several formats like eps, png, jpg.



The pgf package

- ▶ `pgf` stands for **P**ortable **G**raphics **F**ormat
- ▶ used more and more
- ▶ provides a lot of \LaTeX commands for drawing diagrams, pictures etcetera
- ▶ provides more options than the `graphicx` package like **masking** and **interpolation**
- ▶ less powerfull than the `pstricks` drawing package, but it can generate `pdf` directly unlike most other drawing packages

The color package (1)

You can create colored text using the `\textcolor{color}{text}` or by using the `\color{color}` declaration

Example

```
\textcolor{red}{Red} \textcolor{blue}{Blue}  
\color{green}text \color{black}text Red Blue text text
```

The colors black, **white**, red, green, blue, yellow, cyan and magenta are predefined.

The color package (2)

you can mix your own color using

```
\definecolor{name}{colorspace}{mix}
```

Example

```
\definecolor{darkmagenta}{rgb}{.5, 0, .5}
```

creates the color `\textcolor{darkmagenta}{Darkmagenta}`.

creates the color **Darkmagenta**.



More packages

- longtable** for creating really **long tables** that span pages, also in landscape mode...
- pstricks** very **powerful drawing** package
- xypic** also a very **powerful drawing** package
- fancyvrb** for creating **fancy verbatim** text with line numbers, special commands within the verbatim text...
- rotating** for **rotating boxes**, though this may be deprecated...
- acronym** for **creating lists of acronyms** and using them in your text (like bibT_EX but for acronyms)
 - ... there are thousands of packages to be found at <http://www.ctan.org>, go take a look!



\LaTeX Exam portion

- ▶ part of the exam is on \LaTeX .
- ▶ everything in the sheets is part of the material for the exam
- ▶ probably on your computer
- ▶ pure knowledge is not enough, so practise!
- ▶ during the test you will not have the time to look things up