

Herramientas básicas para la investigación

L^AT_EX, an introduction

Dr. Arno Formella

Escola Superior de Enxeñaría Informática
Universidade de Vigo

22/23

- \LaTeX
- Bib \TeX
- pythontex

What a human can read...

- 80 characters per line (= 80 B)
- 60 lines per page (= 5 kB, 0.5 foil)
- 2 pages per foil (= 10 kB, 1 foil)
- 50 foils per day (= 500 kB, 50 foils)
- 400 days per year (= 200 MB, 20 000 foils)
- 50 year per life (= 10 GB, 1 000 000 foils)
- 10 GB of text compresses to roughly 20% (see `dat/...`)
- hence, a memory stick of **2 GB** sufficient for entire life
- 100 m of books in a bookshelf (100 foils per 1 cm).
- **all** can be transmitted in a modern network in 15 s

Binary or ASCII

- all data should be stored in **human readable** format
- with (open) **standard compression** algorithm

- seems that XML has finally made it
- but there is latex, markdown, simple csv, json
(https://en.wikipedia.org/wiki/List_of_file_formats)
- take care of read/write-consistency

Advantages of human readable files

- accessible
- searchable
- formatable
- modifyable
- extendable
- annotable
- portable
- revision controllable

think twice before using binary files

- there are only two reasons to use binary data
 - efficiency due to repeated or random access (use appropriate pre- and post-filters from/to human-readable/binary)
 - you have really binary data (e.g. images, sound, analog...)

You should work with small parts, i.e., modular documents

- reusable units
- small file units
- simpler distributed authoring
- fine grain version control

- a T_EX L^AT_EX, and/or pdfL^AT_EX system
(all files are **human readable**, besides:
executables and compressed font descriptions)
- your `.tex`-file(s)
- other files (like graphics and bibliography) to include

The T_EX System

what is it

- sort of programming language
- works on almost every platform
- output is just the same
- **best** typesetting program around
- you **program** the look-a-like of your document
- open source (since almost 40 years)
- invented and donated by **Donald E. Knuth**
(<https://www-cs-faculty.stanford.edu/~knuth/>)
- macro language written by Leslie Lamport
(<http://lamport.org/>) simplifies use of T_EX hence, L^AT_EX

<https://en.wikibooks.org/wiki/LaTeX/>

- `https://en.wikibooks.org/wiki/LaTeX/Installation`
- on-line web-based, e.g., `https://www.overleaf.com/`

- have an overall idea of the final document
- generate all individual parts (graphics, lists, images, text, references, etc.) either manually or automatically
- run \LaTeX tools to generate final document (including automatically table of contents, table of figures, table of what-so-ever, bibliography, in- and out-text cross-references, indices, glossary, etc.

- first line
 - document class, global settings
- preamble
 - packages to include
 - command and environment definitions
 - layout specifics
- main document
 - all you want to write
 - includes/inputs of other documents

First document

```
\documentclass{article}

\usepackage{times}

\title{Advanced Documentation with \LaTeX}
\author{Arno Formella}

\begin{document}
\maketitle

Hello, my first document...


$$\sin\alpha \cdot \cos\beta$$

\end{document}
```

Advanced Documentation with L^AT_EX

Arno Formella

September 22, 2022

Hello, my first document...

$$\sin \alpha \cdot \cos \beta$$

$$\sin \alpha \cdot \cos \beta$$

- all commands of \LaTeX start with a \backslash
- all environments are surrounded by a correctly nested pair
 $\backslash\text{begin}\{\text{name}\} \dots \backslash\text{end}\{\text{name}\}$
- you can form blocks (same as in C/C++ or Java) using $\{$ and $\}$ to confine actions of commands (i.e., \LaTeX works like a pushdown automaton)
- you can define your own commands and environments

- the classical ones
article, book, letter, report, slides
- some special ones
amsart, amsbook, amsproc, beamer, xatcobeo,
lia, ... etc.
- and journal styles
IEEEtran, sgmlmanual, llncs, egpubl,
elsart, ...

Some packages...

a4wide, afterpage, amsfonts, amssymb, array,
biblatex, caption, color, colortbl, datetime,
dcolumn, enumitem, fancybox, fancyhdr, float,
fontenc, geometry, graphicx, html, hyperref,
ifthen, inputenc, lastpage, listings,
longtable, lscape, makeidx, marginnote,
multicol, multirow, pdfpages, pgf, sectsty,
todonotes, units, url, verbatim

etc.etc.etc.

Document Structure

collection

- more than 1000 commands and environments
- huge set of prepared packages
- everything searchable and documented
(favorite-search-engine, [CTAN](#), tex-distributions)

The internals of a document

- different structure depending on document class
- structural document division in
part, chapter, section, subsection,
subsubsection, paragraph, subparagraph
- automatic (optional) enumeration
- automatic (optional) table of content (and other tables)
- not all categories might be available in a certain document class

Example of a structured document

```
%\documentclass[12pt]{article}
\documentclass[12pt]{book}

\usepackage{times}
\title{Advanced Documentation with \LaTeX}
\author{Arno Formella}
\begin{document}
\maketitle

\part{What's for tea this afternoon...}
Here we go...

\chapter{Chapter}
Here we go...

\section{Section}
Here we go...
\subsection{Subsection}
Here we go...
\subsubsection{Subsubsection}
Here we go...
\paragraph{Paragraph}
Here we go...
\subparagraph{Subparagraph}
Here we go...

\section*{My nice super section two}
\stepcounter{section}
\section{My nice super section three}
\end{document}
```

Example of a description

```
\noindent algo algo algo
```

```
\begin{description}% [labelsep=1em,labelwidth=0.5\textwidth]
% \newlength{\thiswidth}
% \settowidth{\thiswidth}{\widthof{\bfseries description}}
% \setlength{\labelwidth}{\thiswidth}
% \addtolength{\labelwidth}{\labelsep}
% \setlength{\labelsep}{0pt}
\item[description]
% \lipsum[1]
lists simply parts with some type of marks
\item[enumerate]
% \lipsum[1-2]
lists simply parts with some type of marks
\item[itemize]
lists simply parts with some type of marks
\item[list]
the base for general lists
\end{description}
```

```
\noindent algo algo algo
```

Example of an enumeration

algo algo algo

description lists simply parts with some type of marks

enumerate lists simply parts with some type of marks

itemize lists simply parts with some type of marks

list the base for general lists

algo algo algo

Example of a description

```
\begin{enumerate}%[label=\Roman*{.m.}]
\item
  first item, level 0
  \begin{enumerate}
\item first item, level 1
\item second item, level 1
  \begin{enumerate}
\item third item, level 2
  \end{enumerate}
\end{enumerate}
\end{enumerate}
\item
  second item, level 0
\item
  third item, level 0
\end{enumerate}
```


Example of an enumeration

- ① first item, level 0
 - ① first item, level 1
 - ② second item, level 1
 - ① third item, level 2
- ② second item, level 0
- ③ third item, level 0

- you can nest everything
- there exist 4 predefined levels
- there are parameters to control spaces and margins
- there are parameters to modify the symbols

- the font size is defined implicitly
- the effective size depends on the document class
- commands: `tiny`, `small`, etc.
- Huge is the same as huge for 12pt-text

Example

`tiny` `scriptsize` `footnotesize` `small` `normalsize` `large`

Large LARGE huge Huge

Font

Font shape and series

Example

tiny scriptsize footnotesize small normalsize large
Large LARGE huge Huge

Example

tiny scriptsize footnotesize small normalsize large
Large LARGE huge Huge

Example

tiny scriptsize footnotesize small normalsize large
Large LARGE huge Huge

$$P_{rec} = \left| \sqrt{\frac{\epsilon D_0}{2\eta} \cdot \frac{\lambda^2}{4\pi}} \sum_{\forall \Theta, \Phi} E(\Theta, \Phi) \sqrt{F(\Theta, \Phi)} \right|^2$$

\[

```
P_{rec}=
\left|
\sqrt{
\frac{\epsilon D_0}{2\eta} \cdot
\frac{\lambda^2}{4\pi}
}
\sum_{\forall \Theta, \Phi}
E(\Theta, \Phi) \sqrt{F(\Theta, \Phi)}
\right|^2
```

\]

writing more complex formulas

$$\begin{aligned} |E_{\Psi}(\Theta, \Phi)| &= \sqrt{\frac{\eta D_0 \varepsilon P_t}{2\pi}} \sqrt{F(\Theta, \Phi)} \frac{|e^{-jkr}|}{r} \\ &= \sqrt{\frac{2\eta \varepsilon P_t}{\int_0^{2\pi} \int_0^{\pi} F(\Theta, \Phi) \sin \Theta \, d\Theta \, d\Phi}} \\ &\quad \cdot \sqrt{F(\Theta, \Phi)} \frac{|e^{-jkr}|}{r} \end{aligned}$$

writing formulas is easy

$$\sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{x}}}}}}}}$$

```
\[ \sqrt{1+\sqrt{1+\sqrt{1+\sqrt{1+\sqrt{1+\sqrt{1+\sqrt{x}}}}}}}} \]
```

- (almost) all mathematical symbols are available
- different types of characters
(e.g., bold, roman, etc.)
- (almost) all common structures
(e.g., fractions, indices, matrices, roots etc.)
- you can use macros

$$\begin{aligned} & (a_0, a_1, a_2), (b_0, b_1, b_2), (c_0, c_1, c_2) \\ & (A_0, A_1, A_2)_0, (B_0, B_1, B_2)_1, (C_0, C_1, C_2)_2 \\ & (A_{\alpha_0}, A_{\alpha_1}, A_{\alpha_2}), (B_{\beta_0}, B_{\beta_1}, B_{\beta_2}), (C_{\gamma_0}, C_{\gamma_1}, C_{\gamma_2}) \end{aligned}$$

macros for the lazy and the wise

```
\newcommand{\Vector}[1]{  
  ({#1}_0, {#1}_1, {#1}_2)}  
  
\[ \Vector a, \Vector b, \Vector c \  
\[ \Vector A_0, \Vector B_1, \Vector C_2 \  
\[ \Vector{A_\alpha},  
  \Vector{B_\beta},  
  \Vector{C_\gamma} \  
\]
```

try do replicate..

Try to make a \LaTeX -file that reproduces exactly the two page document on the website:

<http://formella.webs.uvigo.es/doc/hbi22/studentwork.pdf>

tables are as easy

this	is	a
table	with	several
rows	and	columns
and	lines	!

```
\begin{center}\begin{tabular}{c|l|r}
this & is & a \\ \hline
table & with & several \\
rows & and & columns \\
and & lines & ! \\
\end{tabular}\end{center}
```

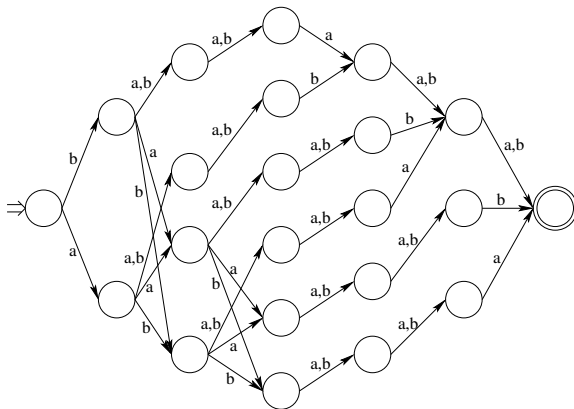
which graphics types can you include

- packages to realize simple graphics directly in \LaTeX
- include graphics in other formats
 - Postscript (EPS) in normal \LaTeX
 - PNG, PDF, and JPEG in pdf\LaTeX

including a PDF figure

```
\centerline{%  
  \includegraphics[width=.7\textwidth]{  
    img/fig_afndln}  
}
```

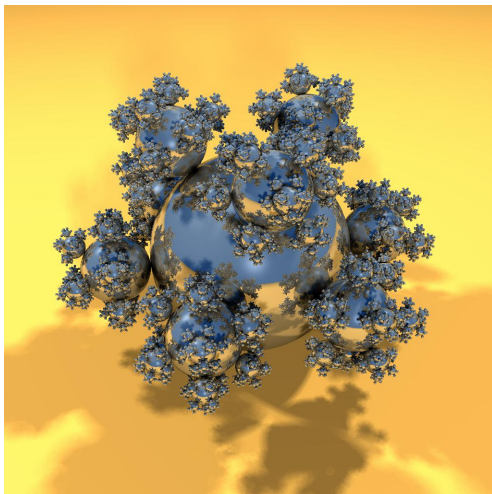
including a PDF figure



including a JPEG image

```
\centerline{ %  
  \includegraphics [width=.6\textwidth] %  
  {img/ballsp}  
}
```


including a JPEG image



Floating objects are automatically placed

- floating objects are figures, tables, images etc.
- \LaTeX puts these objects according to rules (e.g., only at the top of a page)
- permits captions for floating objects
- generates table of figures, table of tables etc.

Floating tables placed where they fit well

this	is	a
table	with	several
rows	and	columns
and	lines	!

Cuadro: A nicely placed table.

Floating tables placed where they fit well

```
\begin{table}[h]
\begin{center}\begin{tabular}{c|l|r}
this & is & a \\ \hline
table & with & several \\
rows & and & columns \\
and & lines & ! \\
\end{tabular}\end{center}
\caption{A nicely placed table.}
\end{table}
```


Embedded floating

```
\begin{floatingfigure}[5.2cm]{%  
\parbox[b]{5cm}{This is an embedded box with floating text  
of with 5cm which automatically aligned  
according to the next paragraph.}  
\end{floatingfigure}%
```

This is some text in a paragraph that will flow around the textbox.

We have to write some more to get the effect,

and more, and more, and more, and more, and more, and more, and more,
and more, and more, and more, and more, and more, and more, and more,
and more, and more, and more, and more, and more, and more, and more,
and more, and more, and more, and more, and more, and more,...

`https://en.wikibooks.org/wiki/LaTeX/Modular_Documents`

What else is there in \LaTeX

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...