

Introduction to \LaTeX

Workshop

Dainius Masiliūnas

Master Geo-Information Science
Wageningen University & Research

April 24, 2017

Table of contents

Theory

Background

Background

Issues with Office

T_EX history and example

Using T_EX

Graphical user interfaces

Collaborative editing

Drawbacks

More information

Questions

Practical

Features

Text and math

Formatting

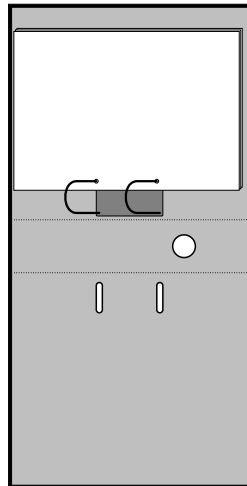
Figures

B_IB_T_EX

CTAN

Background

- Writing reports, papers, theses, articles, ...
- Office text processors: Microsoft Office Word, LibreOffice Writer, WPS Office Words, Calligra Words etc.
- Seemingly good software?
- But has irritations...
- Professionals use $\text{T}_{\text{E}}\text{X}$, a markup language for making documents
- Why?



Issues with Office

- Images: they never stay in place, go above text, outside of caption boxes, etc.
- Copy-pasting text (or worse, tables) leads to formatting problems
- Different office programs render the same files differently
- Bibliography management is a hassle, citation styles are too rigid
- Math formulas handled by another program altogether
- Reformatting (for publishing etc.) is tricky
- Large documents/images become very slow
- \TeX solves that, and more



TEX

- A typesetting system made in 1978 (MS Word: 1990)
 - \LaTeX (1984): additions to base \TeX for ease of use
 - Con \TeX t (1996, Dutch): additions to \TeX for more control
- Document files are plain text (`.tex`, source code, markup)
- An engine (compiler) is used to turn it into a standard read-only document (PDF, PS)
 - pdf \LaTeX
 - X \TeX
- Highly extensible, heavily package-oriented: “there’s a package for that”
- Available as a distribution: collection of engines and packages
- Free and open-source software, cross-platform



Example

```
\documentclass[12pt]{article}
\usepackage{amsmath}
\title{\LaTeX}
\date{}
\begin{document}
\maketitle
\LaTeX{} is a document preparation system for
the \TeX{} typesetting program. It offers
programmable desktop publishing features and
extensive facilities for automating most
aspects of typesetting and desktop publishing ,
including numbering and
cross-referencing ,
tables and figures , page layout ,
bibliographies , and much more.

\LaTeX{} was
originally written in 1984 by Leslie Lamport
and has become the
dominant method for using
\TeX; few people write in plain \TeX{} anymore.
The current version is \LaTeXe.

% This is a comment, not shown in final output.
% The following shows typesetting
power of LaTeX:
\begin{align}
E_0 &= mc^2
\\
E &= \frac{mc^2}{\sqrt{1-\frac{v^2}{c^2}}}
\end{align}

% Same in built-in LaTeX math:
$$ E_0 = mc^2 $$
$$ E = \frac{mc^2}{\sqrt{1-\frac{v^2}{c^2}}} $$
\end{document}
```

$$\text{\LaTeX}$$

\LaTeX is a document preparation system for the \TeX typesetting program. It offers programmable desktop publishing features and extensive facilities for automating most aspects of typesetting and desktop publishing, including numbering and cross-referencing, tables and figures, page layout, bibliographies, and much more.

\LaTeX was originally written in 1984 by Leslie Lamport and has become the dominant method for using \TeX ; few people write in plain \TeX anymore. The current version is \LaTeXe .

$$E_0 = mc^2 \quad (1)$$

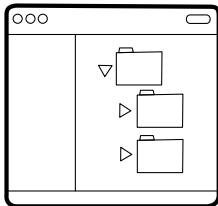
$$E = \frac{mc^2}{\sqrt{1 - \frac{v^2}{c^2}}} \quad (2)$$

$$E_0 = mc^2$$

$$E = \frac{mc^2}{\sqrt{1 - \frac{v^2}{c^2}}}$$

Graphical user interfaces

- Quick access to formatting, math, table creation macros, spell checking, word wrap, search, compiling, viewing the result
- Cross-platform GUIs, such as:
 - [Texmaker](#) and [TeXstudio](#)
 - [Kile](#)
 - [LyX](#)
- All depend on an already installed $\text{T}_{\text{E}}\text{X}$ distribution (i.e. $\text{T}_{\text{E}}\text{X}$ Live) to work!
 - Linux distributions provide $\text{T}_{\text{E}}\text{X}$ Live (and GUIs) out of the box via package managers
 - $\text{T}_{\text{E}}\text{X}$ Live installer for Windows [available](#)
 - macOS can use Mac $\text{T}_{\text{E}}\text{X}$



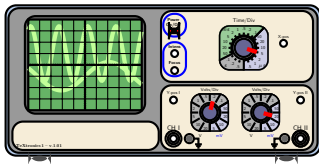
Collaborative editing

- Web-based editors (a la Google Docs) available
 - Papeeria, Authorea, Overleaf, ShareLaTeX, EtherTeX, ...
 - Most require payment for private repositories
- Git (thanks to plain text)
 - GitHub, GitLab, own server, ...
 - Beware of merge conflicts: split chapters into separate files
 - Useful even when writing alone: no more problems with backups and multiple versions



Drawbacks

- Learning curve: pays off in the long run
- Confusing errors (if any!)
 - Auxiliary files in working directory may help: .log for engines, .b1g for BibT_EX
 - Overfull \hboxes: text interpreted as a character/box; if it cannot be broken up nicely, warns you
 - File '<package>.sty' not found. means you don't have the package installed
- Quirkiness:
 - pdf_AT_EX doesn't support UTF-8 symbols
 - Need packages for handling image formats etc.



Want to learn more?

- WikiBooks:
<https://en.wikibooks.org/wiki/LaTeX>
- CTAN: <http://ctan.org>
- Stack Exchange: <http://tex.stackexchange.com/>
- This presentation and examples:
<https://github.com/GreatEmerald/tex-workshop>
- [My Advanced Earth Observation paper](#)
- [My Master thesis and proposal](#)



Questions?

Theory

Background

Background

Issues with Office

T_EX history and example

Using T_EX

Graphical user interfaces

Collaborative editing

Drawbacks

More information

Questions

Practical

Features

Text and math

Formatting

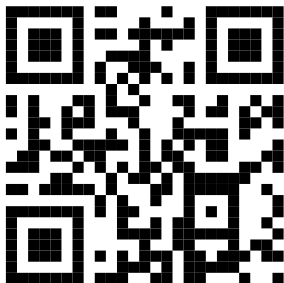
Figures

B_IT_EX

CTAN

Practical

- You can look at, compile and edit the examples in the following slides on Papeeria
- Invitation link: <https://goo.gl/AahZf5>
- Read-only by default; to make your own instance, go to pull-down menu at the top left, “Clone project...”, “Clone”



Text and math (example.tex)

- \TeX is plain text: no formatting is preserved when pasting
- Whitespace is ignored, unless two new lines
- Inline maths capability with math-specific styles, syntax is fairly intuitive

Formatting

(ISPRSGuidelines_authors.tex)

- Formatting can be global or local
 - Global formatting is done in the file header: page margins, line spacing, font size, etc.; geometry package
 - Local formatting is done inline with special commands like `\texttt`, `\textbf`, `\textit`
 - Experts enjoy semantic styles by defining new commands:
`\newcommand{\code}[1]{\texttt{#1}}`
- Journals often provide .cls style templates or template .tex files
 - `<filename>.cls`: use `\documenttype{<filename>}`
 - `<filename>.tex`: possibly use `\input{<filename>}`, edit or use as a reference
 - [Example from Advanced Earth Observation \(ISPRS\)](#)

Figures

(figures.tex)

- \LaTeX has a figure environment that makes it easy to include figures and refer to them
- The position of figures is determined automatically – no problem to add more text
- \TeX is optimised for vector graphics (PDF, EPS, SVG), avoid rasters
- `hyperref` package allows following references within the PDF

BIB_TE_X

(bibliography.tex)

- A system that handles all things bibliography
- Database of bibliography is stored in plain text, .bib files, one may copy-paste entries from Google Scholar
- Cite entries by referring to their aliases
- The default citing package is rigid, use natbib for customisability
- Integrates with hyperref
- Two-step process; might need two compilation rounds

More packages! (`gantt.tex`)

- CTAN: Comprehensive T_EX Archive Network (<http://ctan.org>)
- 5300 packages at the moment of writing! (was: 5199)
 - `pgfgantt`: Gantt charts! Useful for ACT and thesis proposals.
 - `tikz`: flowcharts and other vector graphics!
 - `textcomp`: symbols °, ©, ... for pdfL_AT_EX
 - `qrcode`: QR code generation!
 - `pdfcomment`: comments inside PDF files!
 - `knitr`: embed R code into T_EX!
 - Allows putting the output of *R* commands into plain text, no more copy-pasting results
 - Exact same tool as for `.Rmd`, handles `.Rtex` too
 - Actually an *R* package implementing a T_EX engine
 - `beamer`: Presentations! Like this one!