

LaTeX Typesetting

Anton Gerdelan <gerdelan@scss.tcd.ie>

For LaTeX

- formats professional academic type-set documents
- write in plain text files
 - similar tags to HTML
 - compiled into PDF
 - easy to version in e.g. Git
- can easily split large documents into chapter files
- difficult tasks like TOC and referencing are automatic
- no need to fiddle with fonts or layout styles

MS Word:

$$\iiint_G [u \nabla^2 v + (\nabla u, \nabla v)] d^3 V = \oiint_S \left(u \frac{\partial v}{\partial n} + v \frac{\partial u}{\partial n} \right) d^2 A$$

LaTeX:

$$\iiint_G [u \nabla^2 v - v \nabla^2 u] d^3 V = \oiint_S \left(u \frac{\partial v}{\partial n} - v \frac{\partial u}{\partial n} \right) d^2 A$$

Against LaTeX

- lots of obscure markup tags - comparable to CSS
- compiling document is done in multiple compiler passes
 - figure placement can be a challenge
 - easy to leave unlinked references [?]
- people that use LaTeX for everything
- no Unicode - accents etc require macros

Great For

(and only good for)

- Books
- Thesis
- Reports (**final year reports**)
- Research papers
- Tree assignment instructions where the numbers won't line up properly in a word processor
- CV, formal letters, exam script

Background

- Tex ["teshghhh"] - engine by Donald Knuth (1977)
- LaTeX - macros for Tex by Leslie Lamport (1984+)
- LaTeX2e is the current version (since many years)
- Download styles/packages to pre-format in particular style
 - publishers/confs/orgs usually provide these
 - IEEE, acmsiggraph, Trinity thesis template etc.

Software

- Windows: MiKTeX <http://www.miktex.org>
- Linux:
`apt-get install texlive-latex-recommended`
- Mac: TeXShop
- Desktop install is quite big and a bit clunky so try...
- Online:
 - www.overleaf.com
 - sharelatex.com

Workflow

- screen 1: text editor (or Overleaf)
- screen 2: pdf viewer
- web browser to look up tags/formatting
- probably the best resource:
<http://en.wikibooks.org/wiki/LaTeX/>
- usually start with an example or template `.tex` file and modify it / add your words

Part 1: Create a plain text
file `mydocument.tex`

gnuplotlab.tex

```
%% gnuplot lab worksheet by Anton Gerdelan apg@bth.se
\documentclass{article}

% properties of the title
\author{Anton~Gerdelan}
\date{\today}
\title{Visualisation Lab: gnuplot}

\usepackage{graphicx}

\begin{document}
\maketitle % put the title here

\section{What is gnuplot?}
gnuplot is a tool for creating graphs and charts.

\end{document}
```

Different Tools in LaTeX

- `latex mydoc.tex -> mydoc.dvi`
 - uses .eps images natively
- **pdflatex** `mydoc.tex -> mydoc.pdf`
 - embeds fonts in .pdf but no native .eps support
- `dvipdfm mydoc.dvi -> mydoc.pdf`
 - handy to get a pdf with eps images
- `dvi2ps mydoc.dvi -> mydoc.ps`

Compiler Passes

- Run `pdflatex` once -> it does a **single pass** over document
 - Cross-references e.g. `Figure~\ref{fig:myplot}` are collected in a separate file
 - `.pdf` appears with *Figure ???*
- Run `pdflatex` a second time -> compiles in cross-references
- Citations or glossaries? Run a third pass

Macros and Options

- `\documentclass[twocolumn]{article}`
- LaTeX macros start with backslash
- most macros have optional [options]
- next are curly braces where parameters go
- some characters are reserved, you need a macro to have them appear in the final document
 - # \$ ^ & _ { } ~ \
 - \# \\$ \% \textasciicircum{} \& _ \{ \} \~{}
 - \textbackslash{}
- yeah...

Sections

```
\section{Apparatus}
```

Just start typing in plain text...

Leave a blank line to start a new paragraph (indented in single-column mode).

```
\subsection{Instrumentation}
```

You can type `\textit{italics}` (foreign words), `\textbf{bold face}` for emphasis, or `\texttt{monospace}` for commands.

You can insert algebraic values or mathematics in dollars $n < n^2$.

and for Linux and Windows systems, can usually be found as part of graphics hardware drivers on the manufacturers website. Apple has its own implementation of OpenGL, which installs when the operating system is upgraded. The basic set of libraries comprises `gl` and `glu` (some extra utility functions).

The raw libraries focus only on rendering functionality, but we will adopt some supporting libraries to abstract the context-creation process, which is otherwise quite involved and platform-specific, and to add some extra functionality that we will require. In the past, the 3rd-party GLUT (OpenGL Utility Toolkit) library [8] was commonly used to provide basic input handling, user-interface, time management, and platform-independent context-creation functionality. The library has not been updated for some years, but we can instead use the relatively up-to-date freeGLUT [11] alternative, which implements most of the original GLUT supporting functionality, with some additional improvements.

The most recent OpenGL functionality is usually developed as plug-in **extensions** to the main specifi-

3 OpenGL 4.2 Conventions

Core and compatibility terminology. Binding. - maybe introduce at first bind.

4 Apparatus

Just start typing in plain text...

Leave a blank line to start a new paragraph (indented in single-column).

4.1 Instrumentation

You can type *italics* (foreign words), **bold face** for emphasis, or `monospace` for commands.

You can insert algebraic values or mathematics in dollars $n < n^2$.

Referencing with Bibtex

- create a `mybib.bib` file with all your references {books, articles, papers, etc. }
- each `mybib.bib` entry has a **cite_key**

```
@article{Gerdelan2012,  
  author   = "Gerdelan, A. and Klaus, S.",  
  title    = "Coding Over the Christmas Break",  
  journal  = "{International Journal of Christmas Studies}",  
  year     = "2012"  
}
```

- in your `.tex` preamble add `\usepackage{cite}`
- and in paragraphs "*...has shown that dogs can not look up~\cite{**Gerdelan2012**}.*"
- and just before `\end{document}`

```
\bibliographystyle{plain} % style of references section  
\bibliography{mybib} % name of your file, without .bib extension
```

- structure of `@book @article @inproceedings` etc in LaTeX wikibook

Bullets, numbering

The major advantages of gnuplot are:

```
\begin{itemize}
  \item clean, publication-quality graphs are created
  \item error bars are easy to display
  \item input data is easy to format
  \item data or commands can be piped in from \texttt{stdin}
  \item you have control over every aspect of the display
  \item you can output to a variety of pixel and vector formats
\end{itemize}
```

- `itemize` or `enumerate`
- `item` tag
- the `\texttt{}` tag produces monospace font (code)

Figures

If you want to read about parametric plots you can do something like `Figure~\ref{fig:3dplot}`.

```
\begin{figure}
    \includegraphics[width=1.0\textwidth]{3d.png}
    \caption{a parametric mode ``splot' with hidden3d option
    (depth sorting) disabled \label{fig:3dplot}}
\end{figure}
```

- put `\usepackage{graphicx}` above `\begin{document}`
- LaTeX puts the figure where *it* thinks is best (in my case above the text)
- figure options `[h!tbp]` > options (in order of preference):
"here", "put it here, you bastard!", "top", "bottom", and on a separate page of figures.
- always refer to each figure in the text. `\ref{LABELNAME}`
- tip: writing quotation marks

Tips for Figures

- in two-column try `\includegraphics[width=1.0\columnwidth]{3d.png}`
- use `{figure*}` environment to spread a figure across both columns
- output eps-format with dimensions 0.65", 0.65" for single-column size
- there is a package called `sidewaysfigure` for doing whole-page figures
- put label tags inside caption tags to avoid a really annoying bug
- to compile with `pdflatex` and `.eps` images; `\usepackage{epstopdf}` and:

```
pdflatex -shell-escape myfile.tex
```

Equations

- one of the great advantages of Tex

```
\usepackage{amsmath} % put in preamble
```

```
\begin{equation} \label{eq:someequation}
```

```
f(x) = ax + \frac{bx^{32t}}{\Delta \epsilon}
```

```
\pi = \sqrt{12} \sum^{\infty}_{k=0} \frac{(-3)^{-k}}{2k+1}
```

```
\end{equation}
```

```
...as described in equation~\ref{eq:someequation}..
```

$$f(x) = ax + \frac{bx^{32t}}{\Delta\epsilon}\pi = \sqrt{12} \sum_{k=0}^{\infty} \frac{(-3)^{-k}}{2k+1} \quad (1)$$

...as described in equation 1.

Tables

- use the tabular environment
- arguments give left, right, centre justification of columns, and | or || for dividers
- columns are separated by &
- to get a new row use \\
- \hline draws a line under a row

```
\begin{tabular}{ l c r }  
 1 & 2 & 3 \\  
 4 & 5 & 6 \\  
 7 & 8 & 9 \\  
\end{tabular}
```

Better Tables

- put tabular inside a table environment for caption and label
- table environment works a lot like figure environment (same `h t b p !` options)

```
\begin[h!tbp]{table}
  \caption{Performance at peak F-measure \label{tab:fmeasure}}
  \begin{tabular}{| r | r || c | c | c |}

      ...

  \end{tabular}
\end{table}
```

- hint: use minimal lines; usually just a bold row of labels is enough
- again: refer to each table in the text with `\ref{LABEL}`, and pdflatex twice

Errors and Warnings

- type **x** to quit, enter to continue, or q to be quiet and carry on
- usually missing packages; should be auto-installed with MiKTeX
- maths symbols must be inside **\$\$**
- also check the bibtex log to see if there are any mistakes in the .bib or missing references
- descriptions of common errors:
http://en.wikibooks.org/wiki/LaTeX/Errors_and_Warnings

Fancy Things

- Various tools for building/managing .bib file
- `\include{chapter2.tex}` to include other files
- code fragments with ***listings*** package
- algorithms with ***algorithmic***
- tools like gnuplot to build professional-looking charts
- changing fonts/styles/classes
- adding and removing vertical whitespace
- circuit diagrams/music notation/chemistry diagrams
- "beamer" for presentation slides [never again]

Typical Article Structure

- title, authors, institute, emails, date
 - abstract (use built-in style for this) - **include spoilers / major outcome**
1. introduction
 - literature review (thesis) / previous works (paper) - **lots of citations**
 - state of the art or latest developments in area
 - aim/hypothesis/"research question" [eww!]
 2. experiment design or method
 3. results
 4. discussion/conclusion/future work
- references