

Latex & Markdown

Tópicos de Informática para Automação

Mário Antunes

December 8, 2025

Universidade de Aveiro

Table of Contents i

LaTeX & Markdown

LaTeX

Markdown

ToC

Figures

Tables

Bibliography

Mathematical Equations

Further Resources

LaTeX & Markdown

Document Generation Paradigms i

1. WYSIWYG (What You See Is What You Get)

- **Examples:** Microsoft Word, Google Docs, LibreOffice Writer.
- **Concept:** The editing interface is a mirror of the final print output.
- **Mechanism:** Direct manipulation. Formatting (bold, size, font) is applied directly to text characters.
- **Pros:** Low barrier to entry; immediate visual feedback.
- **Cons:** “What You See Is All You’ve Got.”
 - Content is tightly coupled with presentation.
 - Moving a generic “Header” to a “Chapter Title” requires manual reformatting.
 - Complex documents (theses, books) often break formatting when moved between computers or versions.

2. WYSIWYM (What You See Is What You Mean)

- **Examples:** LaTeX, Markdown, HTML, AsciiDoc.
- **Concept:** You edit the **semantic structure** (meaning), a compiler handles the **visual presentation**.
- **Mechanism:** Separation of concerns.
 - *Source:* Plain text (.tex, .md) containing content and logical tags (e.g., \section, # Heading).
 - *Engine:* A compiler (e.g., pdflatex, pandoc) applies a specific style template to generate the output (PDF, HTML, EPUB).
- **Pros:** Consistency, automation, superior typography.

The Value of Compiled Documents i

Why learn a complex syntax when Word exists?

A. Repository Usage (Git)

- **Text vs. Binary:** Word files (.docx) are zipped XML binaries. Version control systems (Git) treat them as “blobs.” You cannot meaningfully diff them.
- **Line-by-Line Tracking:** In LaTeX/Markdown, you can track changes down to the specific character and see the commit history.
- **Branching:** Ideal for experimenting with a new chapter structure without breaking the main document.

B. Collaboration

- **No “File Locked” Issues:** Unlike opening a .docx on a network drive, multiple people can edit different text files in a project simultaneously.
- **Merging:** Git allows merging different text files automatically.
- **Modularity:** Large documents are split using `\input{chapter1.tex}`, keeping files small and manageable.

C. Migration & Templates

- **Abstraction:** In WYSIWYM, you tag text as “Title” or “Abstract”. You do not choose font size or margins manually.
- **Content Portability:**
 - *Scenario:* You write a thesis. Later, you want to publish a chapter as a conference paper.
 - *Action:* Change `\documentclass{thesis}` to `\documentclass{ieee-conf}`.
 - *Result:* The entire document is reformatted (fonts, columns, citations) instantly. No manual reformatting required.

LaTeX

Deep Dive: LaTeX i

LaTeX is the industry standard for scientific and technical communication. It is practically Turing-complete.

Document Structure

A LaTeX file (`.tex`) has two distinct parts:

1. **The Preamble:** Everything *before* `\begin{document}`.
 - Defines the “Class” (style).
 - Loads “Packages” (plugins for extra features like images, colors, links).
 - Sets global parameters (margins, meta-data).
2. **The Body:** The content environment inside `\begin{document} ... \end{document}`.

Key Entities

- **Classes:**
 - `article`: Scientific papers, short reports.
 - `report`: Longer documents with chapters (theses).
 - `book`: Books with front/back matter support.
 - `beamer`: For creating presentation slides.
- **Environments:**
 - Blocks of logic defined by `\begin{name}` ... `\end{name}`.
 - *Examples:* `itemize` (lists), `equation` (math), `tabular` (tables), `center`.
- **Floats (Figures & Tables):**
 - LaTeX decides where to place images (`\begin{figure}`) for optimal reading flow.

Usage & Compilation

- **CLI:** `pdflatex main.tex` (Single pass).
- **The 3-Pass Rule:** Often requires running 3 times to sync references (Pass 1: Collect labels; Pass 2: Assign numbers; Pass 3: Fix layout).
- **Automation:** `latexmk -pdf -pvc main.tex` (Watches for file changes, handles cross-references automatically).
- **Cloud: Overleaf.** A browser-based editor that manages the compiler installation for you.

Markdown

Deep Dive: Markdown i

Markdown is a lightweight markup language designed for readability.

Philosophy & Syntax

The goal is that the raw source file should be readable as plain text without looking like computer code.

- **Headers:** # for H1, ## for H2 (translates to <h1>, <h2>).
- **Lists:** - or * for bullets; 1. for numbered.
- **Formatting:** **Bold** (), *Italic* (<i>), ``Code``.
- **Links:** [Text](URL).
- **Images:** ![Alt Text](URL).

Flavors & Extensions

Markdown has evolved into several “Flavors”:

- **CommonMark:** The standardized specification.
- **GFM (GitHub Flavored Markdown):** Adds tables, task lists (– []), and strikethrough.
- **Pandoc Markdown:** The most powerful version. Adds citations (@author), footnotes ([^1]), metadata blocks, and math support.

Usage (Pandoc)

Pandoc is the “Universal Converter”. It reads Markdown and outputs almost anything.

- **Logic:** Markdown → Abstract Syntax Tree (AST) → Output Format.
- **Commands:**
 - `pandoc input.md -o output.pdf` (Uses LaTeX engine).
 - `pandoc input.md -o output.docx` (Generates Word).
 - `pandoc input.md -t beamer -o slides.pdf` (Generates LaTeX slides).

ToC

LaTeX ToC

LaTeX generates a ToC automatically by scanning your Section tags (`\section`, `\subsection`).

- **The Command:** Simply place `\tableofcontents` where you want the list.
- **Compilation Mechanism:**
 1. *Run 1:* LaTeX writes all section titles and page numbers to a temporary `.toc` file.
 2. *Run 2:* LaTeX reads the `.toc` file and renders the list in the document.

- **Example:**

```
\begin{document}  
  \maketitle  
  \tableofcontents % Auto-generates here  
  \newpage  
  \section{Introduction}  
\end{document}
```

Markdown ToC

Standard Markdown does not have a strict ToC tag, but tools handle it differently:

1. **Pandoc:** Use the flag `--toc` in the command line. It scans headers (`#`, `##`) to build it.
2. **Editors (VS Code/Typora):** Many support the macro `[TOC]`.
3. **Manual:** You write it as a list of links: `- [Introduction](#introduction)`.

Figures

The `graphicx` Package

LaTeX treats images as “floats”—it automatically decides the best position (top of page, bottom, etc.) to prevent awkward page breaks.

- **Prerequisite:** You must add `\usepackage{graphicx}` to your preamble.
- **Environment:** `\begin{figure}[placement]`.
 - *Placement options:* h (here), t (top), b (bottom), ! (override constraints).

- **Key Commands:**

- `\includegraphics[options]{filename}`: The actual image insertion.
- `\caption{ ... }`: Adds the description and numbering (e.g., "Figure 1: ...").
- `\label{ ... }`: Creates an anchor to reference it later (e.g., "See Figure `\ref{...}`").

LaTeX Code:

```
\begin{figure}[ht]
  \centering
  \includegraphics[width=0.5\textwidth]{results.png}
  \caption{Experimental Results}
  \label{fig:results}
\end{figure}
```


Inline Images

Markdown syntax is concise (`![]()`) but typically places images **inline** (exactly where you type them) rather than floating them. Standard Markdown lacks native resizing, but extensions handle this.

1. Standard Syntax: `![Alt Text for Accessibility](path/to/image.png)`

2. With Resizing (HTML Fallback): Since Markdown supports raw HTML, this is the most compatible method for resizing. ``

3. With Resizing (Pandoc Extension): If using Pandoc (standard for academic writing), you can use attributes.

```
![Results](image.png){ width=50% }
```

4. Adding Captions: In Pandoc, the “Alt Text” (text inside []) automatically becomes the Figure Caption below the image when converting to PDF/LaTeX.

Tables

LaTeX Tables (`tabular`)

LaTeX tables are precise but verbose. They use specific delimiters.

- **Environment:** `\begin{tabular}{cols}`
- **Column Spec:** `{l c r}` defines 3 columns (Left aligned, Centered, Right aligned).
- **Separators:** `&` separates cells; `\\` ends a row.
- **Lines:** `\hline` draws horizontal lines; `|` in column spec draws vertical lines.

Tables ii

LaTeX Code:

```
\begin{table}[h]
  \centering
  \begin{tabular}{|l|c|r|}
    \hline
    \textbf{Item} & \textbf{Qty} & \textbf{Price} \\
    \hline
    Apples & 5 & \$1.00 \\
    Oranges & 10 & \$2.50 \\
    \hline
  \end{tabular}
  \caption{Grocery List}
\end{table}
```

Markdown Tables

Markdown uses “ASCII Art” style pipes. It is simpler and easier to read in raw code, but less flexible (no merged cells or complex alignment).

Markdown Code:

Item	Qty	Price
Apples	5	\$1.00
Oranges	10	\$2.50

<-- Alignment (Left, Center, Right)

Bibliography

LaTeX (BibTeX / BibLaTeX)

Citations are stored in a separate plain text database file (.bib).

1. The Database (refs.bib):

```
@article{einstein1905,  
  author = "Albert Einstein",  
  title = "On the Electrodynamics of Moving Bodies",  
  year = "1905"  
}
```


2. The Document:

As stated by `\cite{einstein1905}`, relativity is complex.

```
\bibliographystyle{plain}  
\bibliography{refs}
```

Markdown (Pandoc Citeproc)

Pandoc can read BibTeX files and process citations in Markdown.

The Syntax:

As stated by [Einstein1905], relativity is complex.

The Command: `pandoc doc.md
--bibliography=refs.bib --citeproc -o doc.pdf`

In academic papers (especially IEEE), “Biographies” are formatted blocks at the end of a paper containing the author’s photo and a short bio.

LaTeX (IEEEtran Class)

The IEEEtran class provides a specific environment for this. It handles wrapping the text around the photo automatically and styling the name in bold capital letters.

Biography Generation ii

LaTeX Code:

```
% Requires \documentclass{IEEEtran}
```

```
\begin{IEEEbiography}[{\includegraphics[width=1in,clip,  
received the B.S. degree in aerospace engineering ...  
He is currently a Professor at X University.  
His research interests include LaTeX and Typography.  
\end{IEEEbiography}
```

Biography Generation iii

Markdown

Markdown does not have a native “Biography” semantic tag. You create it manually using Headers and Images, or HTML if rendering to web.

Markdown Code:

`## Author Biography`

`![[John Doe](photo.jpg){ width=100px align=left }`

`**John Doe** received the B.S. degree in
aerospace engineering ...
He is currently a Professor at X University.`

Mathematical Equations

Mathematical Equations i

One of the primary reasons to use WYSIWYM is the superior rendering of mathematics.

LaTeX Math

LaTeX has two modes:

1. **Inline Mode:** For math inside a sentence. Surrounded by $$.
 - *Syntax:* Let x be a variable.$
2. **Display Mode:** For centered, standalone equations.
 - *Syntax:*
$$E = mc^2$$
 or $\begin{equation} \dots \end{equation}$.

Mathematical Equations ii

Common Commands:

- **Fractions:** `\frac{numerator}{denominator}`
- **Greek:** `\alpha`, `\beta`, `\Omega`
- **Summation/Integrals:** `\sum_{i=0}^n`,
`\int_0^{\infty}`
- **Sub/Superscripts:** `x_i`, `x^2`

Mathematical Equations iii

Markdown Math

Most Markdown engines (GitHub, Pandoc, Obsidian, Jupyter) use **MathJax** or **KaTeX** to render LaTeX syntax inside Markdown.

- **Syntax:** It generally uses the exact same $\$$ delimiters as LaTeX.
 - Inline: $E=mc^2$
 - Block:
$$E=mc^2$$

Comparison Example (Quadratic Formula): Both LaTeX and Markdown use: $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

Further Resources

LaTeX

- **Overleaf Learn:** <https://www.overleaf.com/learn> (The best beginner-friendly documentation).
- **CTAN (Comprehensive TeX Archive Network):** <https://ctan.org/> (The central repository for all LaTeX packages).
- **Detexify:** <https://detexify.kirelabs.org/classify.html> (Draw a symbol to find its LaTeX command).

Markdown & Pandoc

- **Markdown Guide:** <https://www.markdownguide.org/>
(Comprehensive tutorial on syntax and flavors).
- **Pandoc Documentation:** <https://pandoc.org/> (The manual for the universal converter).
- **GitHub Flavored Markdown Spec:**
<https://github.github.com/gfm/>

Tools

- **Editors:** VS Code (with LaTeX Workshop & Markdown All in One).
- **Reference Management:** Jabref (Works directly with BibTeX).