Typst, a new tool for writing scientific documents

Baptiste Pesquet

November 17, 2026







Outline

Why Typst? 2

- Scientists: text is all you need 😀 3
- What's in a scientific document 4
- The king on the hill: (La)TeX 5
- The new kid on the block: Typst 6

Features 7

- Principles and syntax 8
- Demo time! 👓 9

Should you use Typst? 10

- Pros 11
- Cons 12

Why Typst?

Scientists: text is all you need 😀

- For centuries, *text* has been the dominant medium for communicating scientific ideas.
- Main usages:
  - Publishing results through articles and reports.
  - Exchanging information with colleagues
- Why?
  - Universal
  - Precise
  - Easy to index, search and archivate

What's in a scientific document

- Textual content
- Annotated figures
- Mathematical equations
- Algorithms
- Citations and bibliographies
- Footnotes
- And more!

The king on the hill: (La)TeX

- TeX is a typesetting system created en 1978 by Donald Knuth.
- LaTeX is a set of macros designed in 1984 by Leslie Lamport to facilitate the usage of TeX.
  - Current version: LaTeX2e (1994)
  - LaTeX3 is in development since 1989
- Separate content from rendering, purely text-based, free software licence, rich ecosystem.
- Gold standard for authoring and publishing scientific material.

The new kid on the block: Typst

- Typst is a new technology stack for creating scientific documents.
  - First public release: 2023
- Provides:
  - An open source CLI compiler.
  - A collaborative online editor similar in spirit to LaTeX's Overleaf.

Features

Principles and syntax

- Designed to be “as powerful as LaTeX while being much easier to learn and use”.
- Based on the Markdown syntax.
- Three syntactical modes:
  - markup (the default)
  - math
  - code (for scripting)
- Compiler built in Rust.

Demo time! 👓

- Examples:
  - Syntax basics [source | result]
  - Math [source | result]
  - Article draft [source | result]
  - Slideshow (this doc!) [source]. Uses Touying
- See the official tutorial for more details.

Should you use Typst?

Pros

- Syntax clarity
- Speed!
- No need for complex package management
- User-friendly error messages
- Growing ecosystem (themes, extensions, etc)
- Positive returns by early adopters

Cons

- Format not accepted by conferences and journals ⛔
  - Should change sooner or later
- Tiny community compared to LaTeX's
  - Less support and resources
  - Need for home-made customization in some scenarios
- Uncertainty about the project's future (planned roadmap)

Thank you for your attention!

Any questions? 😀