About Typst

Richard Koch

October 16, 2023

1 Introduction

There are several projects in the TeX world attempting to rewrite TEX from scratch. Sometimes these programs input LATEX source files and just modify the typesetting code. LuaTEX and XeTEX are familiar examples. Other programs start completely from scratch with new markup input languages. Typst is one of these.

Another interesting example is JSBox by Doug McKenna. This was described at TUG meetings in 2014 and 2019. One goal of the project was to create documents for iPhones and iPads which reflow immediately when the user resizes the window, and which can contain interactive illustrations. JSBox is not well-known because McKenna did not release it as open source, but the iOS app *Hilbert Curves* by McKenna shows many of its capabilities. In his 2014 report, McKenna discussed showing the program to Donald Knuth, who encouraged him but warned that "any rewrite of TEX will be a full time job taking at least five years."

More recently Martin Haug and Laurenz Mädje began a project in Berlin to rewrite both the input language and the typesetting code for a Latex-like program. Their program is named *Typst* and is programmed in Rust. The project began in 2019, so Knuth's estimate of the time it might take remains quite accurate. Details about the reasons for the rewrite, and the goals, can be found at their web site `https://typst.app`.

2 A Typst Engine

Recently I received a TeXShop engine file from Jeroen Scheerder which can typeset Typst source files. These source files usually have extension ".typ", so I added that as a file type which TeXShop recognizes and is willing to typeset. Thus TeXShop users can easily experiment with the new typesetting engine and its distinctive input language.

To set this up, perform the following steps:

- Go to https://github.com/typst/typst/releases/, scroll down to the “Assets” section, and download either typst-aarch64-apple-darwin.tar.xz or typst-x86_64-apple-darwin.tar.xz depending on whether you have an Arm processor or an Intel processor. The zip file will decompress into a folder containing “typst” and some license and readme files. The file “typst” is the full typesetting program.
- If you try to run typst, the Mac will display a dialog reading

  ```
  "typst" can’t be opened because Apple
  cannot check it for malicious software.
  ```

  We should ask the authors to notarize the file with Apple, but if you trust them, you can remove the warning by opening Terminal, changing to the directory containing typst, and typing

  ```
  xattr -d com.apple.quarantine typst
  ```

- Then drag typst to /usr/local/bin.
- Find the file Typst.engine in the folder containing this document and drag a copy to the active engine folder, /Library/TeXShop/Engines.

Now you are ready to experiment. If you are given a Typst source file, add the following line to the top of the file

```
//% !TEX TS-program = Typst
```

This line tells TeXShop to typeset using Typst. Unfortunately, % is not a comment symbol in the Typst input language, so we preface the line with the comment symbol in that language, //. It is not necessary to remember this line; just choose the TeXShop Macro titled “Program” and a list of active typesetting engines will appear. Select ”Typst” and the line will be written at the top of your source. Add the extra // at the beginning.

If you start a new source file completely from scratch, TeXShop will display a Save Dialog the first time you typeset it. The file must be saved with extension “.typ” so Typst will recognize it. It is tempting to just type the new extension when you name the file, but that will not work because TeXShop will add an extra ”.tex” to the end of the filename when saving. Instead, find the pulldown menu ”File Format:” at the bottom of the dialog and select ”typ”; it is the very last element in the menu. Once the file has been saved with the proper extension, TeXShop will use that extension from then on.

Recall that TeXShop has a ”Templates” menu in the source toolbar, and new templates can be added to this item by simply adding their files to /Library/TeXShop/Templates. Until this version of TeXShop, these template files needed to have extension ”.tex”. Now files with extension ”.typ” are also allowed there.

3 Sample Source Files

Since Typst introduces an entirely new input language, it would be desirable to obtain sources for several substantial documents to show what the program can do and how using it differs from using standard LaTeX. As a mathematician, for instance, I'd like to see the source for a 50 to 100 page set of lecture notes, complete will illustrations, complicated inline and displayed equations, tables, commutative diagrams, and the like.

Unfortunately, such source documents do not seem to be available yet, although providing them is on the author's to-do list. Instead the Typst web page contains an elaborate manual for using the program, with many snippets explaining items that are easier to input in Typst than in LaTeX. These snippets give a glimpse of what is possible, but close study of the manual is required to see how they all fit together.

In the meantime, however, I discovered a folder on the Typst site, with five samples titled *ams, dept-news, fiction, ieee,* and *letter.* Each of these produces a complete document and these documents give a much clearer view of the current capabilities of the program. See the folder SamplePrograms, with its five subfolders. In each subfolder, open and typeset the file *main.typ.* The magic line telling TeXShop to typeset with Typst has already been added to these files.

4 Acknowledgement

This new capability is really the work of Jeroen Scheerder, who wrote the engine file. Without his encouragement, I would not have looked at the Typst site, and certainly would not have realized that the project is very approachable in its current state. Contact Scheerder at Jeroen Scheerder <js@gumby.nl> if you have questions about the engine.

5 Packages

After the Typst folder was added in TeXShop version 5.20, Scheerder created a simple modification making it easier to use. This modification is based on *Packages.* Packages are the rough equivalent of sty and class files in the LaTeX world; a Typst source document can input package files to extend the capabilities of the language. An elaborate package system for Typst is under construction currently. See `https://github.com/typst/packages#local-packages` for details.

In particular, the Typst world has an equivalent of ~/Library/texmf where users can store their own packages and experimental packages from others not yet published for general use. On the Macintosh, this location is ~/Library/Application Support/typst/packages. This location takes precedence over others, just as ~/Library/texmf is searched first in LaTeX.

Scheerder's modification split each sample program discussed earlier into two pieces, a package which any source file can use, and a template which can be used to start an appropriate source. The advantage is that users can modify the templates for their individual needs without having to understand or touch the underlying packages.

The modifications are in the folder "Advanced" in the same location as the file you are reading. Installing them is easy. Open ~/Library/Application Support. Create a subfolder named "typst", and a subfolder of that named "Packages". Drag or copy the folder named "typst" inside "Advanced" into the Packages folder. Then drag or copy the folder named "TypstTemplates" inside "Advanced" into ~/Library/TeXShop/Templates. Done.

Now suppose you want to write a letter using Typst. Open a new source window in TeXShop. Find the Templates pulldown in the source toolbar, and notice that it now contains a folder named TypstTemplates. Select the template "letter" in this folder and the source for a letter will appear in your new source window. Revise the text as you wish. Then typeset. In the resulting Save Dialog, be sure to use the FileFormat: menu at the bottom to select file type "typst".

Recall that TeXShop templates can be edited. If the boilerplate text in the "letter" template is annoying, you can easily remove it and create a template which only contains the skeleton needed for each new letter.

This process works for all other templates in the TypstTemplates folder except the ams template. That template has one minor problem due to the fact that the design of Packages is not yet finished. The ams Package inputs a file named refs.bib with a sample bibliography. However, refs.bib should not be part of the package; instead it should be part of the source folder being created by the user writing a new article, because the references change for each article. Scheerder modified the ams template slightly so it looks for refs.bib in the source folder. But if that file is missing in the source file, then typesetting stops with an error.

The easy solution is to add a file named refs.bib to the source folder for any ams article you write. The folder Advanced/TypstTemplates contains such a refs.bib file. Copy it to your first ams source folder. TeXShop can open and edit this file, so it is easy to convert it to a bibliography for your own document.. Any time you create a new file using the ams template, find a copy of this file in one of your other projects and copy it to the new project folder.