
L^AT_EX and Jupyter, TikZ and Vega

Jonathan Fine

1 Then and now

When Don Knuth created T_EX in the 1970s and 80s, publishing was mostly on paper. T_EX was created to solve the problem of computer typesetting, particularly for technical content. The portable computers, including the mobile phone, have changed publishing. Many people prefer laptop and notebook computers to paper books.

2 Laboratory and scientific notebooks

The great experimental physicist Michael Faraday (1791–1867) kept a lab diary. Today we might do this on a computer, as a private blog, or a scientific notebook, such as Jupyter.

T_EX and L^AT_EX solved the problem of typesetting, for printing on paper. Today, Project Jupyter develops “open-source software, open standards, and services for interactive and reproducible computing”.

3 Jupyter and L^AT_EX

In many ways, Jupyter is now what L^AT_EX was in the 1980s. It’s got a growing and well-funded community, and making steady and rapid progress. It is a major and well-respected force.

4 PGF/TikZ and D3/Vega

PGF/TikZ is a deservedly popular T_EX-based technical drawing package. In it, PGF/TikZ is a low-level/high-level language pair.

In the parallel universe of scientific web publishing, D3/Vega is a similar language pair, based not on T_EX but on HTML5.

Many would benefit from a bridge between TikZ and Vega, particularly those who want high-quality visualisation in both PDF and interactive HTML5.

5 Further reading (and browsing)

In January 2018 Nature published a Toolbox article *Data visualization tools drive interactivity and reproducibility in online publishing*. The URL is <https://www.nature.com/articles/d41586-018-01322-9>.

Inspired by this Nature article, I gave a talk at the March 2018 London PyData meetup. The URL is <https://jfine2358.github.io/slides/2018-nature-jupyter-altair-vega-binder.html>.

◇ Jonathan Fine
Milton Keynes
England
jfine2358@gmail.com