TEXShop Review

Abstract

This paper is an introduction to and review of the TEXShop Mac OS X program for type-setting TEX document source files and previewing the output. Features of this tool and user experience will be presented. This information might be helpful to the novice user, looking for a TEX typesetting environment on the Mac.

Introduction

For some decades now I have been using plain TEX and LATEX computer typesetting for scientific documents, lecture notes and other stuff. After starting typesetting on DEC VAX/VMS computer terminals I was pleased to see the integrated development environments for both PC/Windows and Mac OS.

Currently, I use the TEXshop Version 1.42, installed with the teTEX distribution on an old Mac PowerBook G4. My office PC runs a MikTEX distribution with TEXnicCentre (discussed elsewhere in this Maps edition). My first impression was that TEXShop is fairly basic, but reliable.

The next section will describe a non-exhaustive number of features, included in TEXShop, and report user experience.

TEXShop features

This section list essential features of TEXShop, with some user comments added. In a typical typesetting session the program will open three windows, as shown in Figure 1: the source editor, the PDF preview and the console window that report the document compilation progress, warnings and errors.
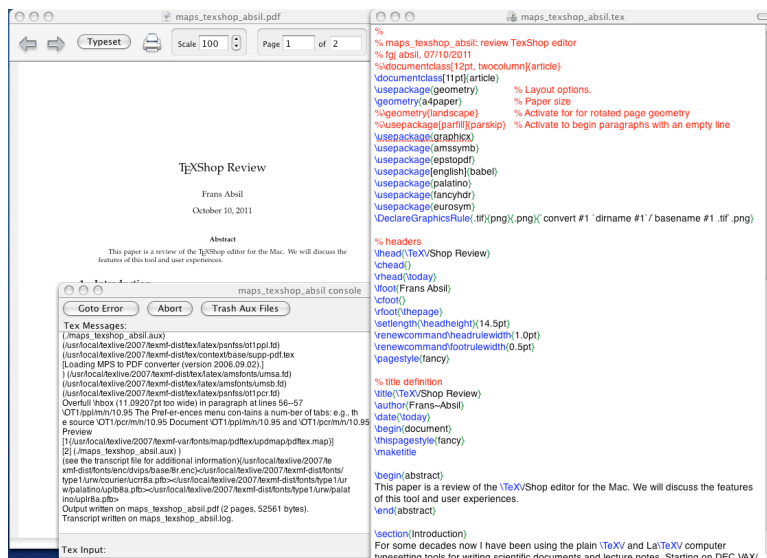

Figure 1. Overview of the three TEXshop windows: editor, previewer and console
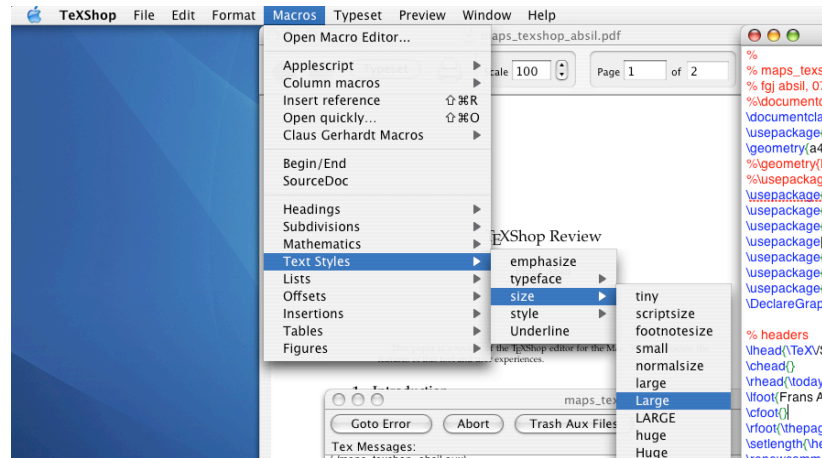
Figure 2. TEXshop **Format** menu with font selection items

Preferences and typesetting engine

The **Preferences** menu contains a number of tabs: e.g., the source `Document` editor font setting and window position, `Preview` window parameter settings, the typesetting `Engine` binary path and and options lists. The default typesetting engine is tickmarked in the `Typesetting` tab. The full set of engines, including TEX, LATEX, ConTEXt, XeLATEX is available in the pulldown menu `Typeset`, where also workflow enhancement scripts or tools such as Pdftex, TEX+Ghostscript, BibTEX and MakeIndex can be found.

As a LATEX user I select either the Pdftex or TEX+Ghostscript typesetting engine; the latter is required for including scientific diagrams created with the `PStricks` package. Obviously in that case all figures have to be available as separate PostScript `.eps` files, whereas for the former option all figures are included as PDF files. And, by the way, drop recognized figure filetypes on the editor window, and it will generate the appropriate `\includegraphics` command.

Editor features

The source file editor uses colour coding, with LATEX commands shown in blue, comments in red, and parenthesis grouping in green, for easy code consistency checks. Although line numbers are not displayed, the **Edit** menu contains a `Line Number` and `Go to Error` command. This menu also contains items for running a spell checker and showing document statistics, such as word, line and character count excluding the LATEX commands.

The **Find** panel, under the **Window** menu, has a number of nifty options: it accepts regular expressions for advanced search, it applies find and replace to selected subregions in the source document (local scope) and it will list all appearances of the search string, a convenient option for navigation and as check before a global replace. It will also remember all previous search and replace strings for the current session.

Panels for the novice user

LATEX commands are mnemonic. However, at the entry level TEXshop contains menu items and special panels for entering source code. Figure 2 shows the menu for font selection. A separate LATEX panel (see Figure 3) shows the most frequently used symbols, environments and other document elements. Clicking an item on the panel will put the LATEX source code in the current document in the editor. TEXshop allows multiple editor windows.

In the **Preview** window there are the usual scrolling, paging and scaling functions. A nice feature is the magnifying glass; when selected clicking anywhere in
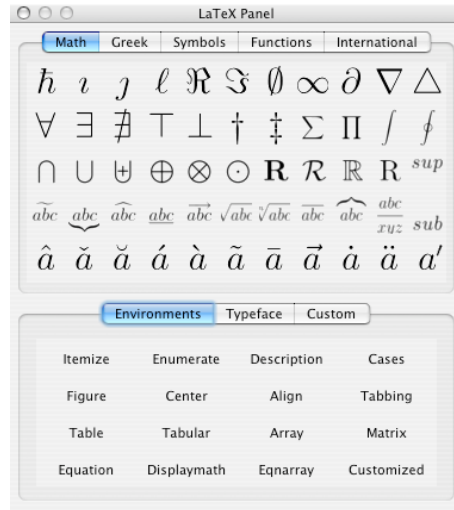
Figure 3. LATEX panel for selection of common symbols and
document elements.

the previewer document will show that section in sufficient magnification to inspect
typesetting details.

The novice user might also refer to the concise but adequate **Help** menu.

Miscellaneous

The **Format** menu has handy items for (un)commenting or indenting blocks of
source code, a feature that is useful when debugging documents.

The **Macros** menu contains items for automating the workflow. My favourite from
this set is the Insert reference: it opens a window with a list of \label commands
in the current document for selecting the appropriate entry. Also convenient is the
Bibliography Applescript, which does multiple runs for creating bibliographical ref-
erences in a document.

For larger documents, that consist of a set of smaller files (e.g., book chapters),
there is the option of setting the project root, i.e., the path to the project main.tex
file; it is available under the **File** menu. What is missing however, is a window rep-
resenting the structure of a large-scale project as a tree graph with directory paths
to source files, figures etc. TEX/nicCentre has this window that is a great help during
editing and debugging report or book class documents.

Conclusion

The old version of TEXShop, running on a Mac OS 10.3.8 operating system, is a re-
liable workhorse. Although there are differences I have no problem using this tool
and TEXnicCentre on the Windows PC in parallel.

A much more recent version, i.e., TEXShop, vs. 3.11 for Mac OS 10.7 (Lion), is
available at http://www.uoregon.edu/~koch/texshop/, and on the TEXLive DVD. A
comparison between various LATEX editors can be found on http://en.wikipedia.org,
containing a table with the features of each.

Frans Absil