

ABOUT TEXSHOP 2.33

RICHARD KOCH

1. THIS DOCUMENT

If you are a new TeXShop user, you can skip this document and begin using the program.

The default behavior of TeXShop can be modified in two ways: by making Preference changes, and by editing files in `~/Library/TeXShop`. Here `~/Library` is the Library folder in your home directory.

New versions of TeXShop generally do not change these user modifications because users don't like changes made behind their backs. But sometimes, new features require a few modifications. This paper explains what to do.

In the future, please read this document when you upgrade. Each release of TeXShop will contain a new version. This version explains changes for TeXShop 2.30 – 2.33.

2. PREFERENCE CHANGES; NO NEW CHANGES AFTER 2.30

TeXShop 2.30 can find errors in projects governed by a root document. When the user selects “Goto Error,” the source file containing the error is opened and the line with the error is highlighted. TeXShop finds errors by parsing the console output. In the default operation of TeX, the console error message does not identify the source file, but this can be changed if TeX is run with the “file-line-error” flag. Therefore, the following Preference changes are needed under the Engine tab. If you modified TeXShop's defaults earlier, for instance by omitting the shell-escape flag, modify the suggestions below appropriately.

Under the engine tab, the pdfTeX item should be

```
pdftex --file-line-error --shell-escape --synctex=1
```

the pdfLaTeX item should be

```
pdflatex --file-line-error --shell-escape --synctex=1
```

the TeX item for TeX + dvips + distiller should be

```
simpdftex etex --maxpfb --extratexopts "-file-line-error -synctex=1"
```

and the corresponding Latex item should be

```
simpdftex latex --maxpfb --extratexopts "-file-line-error -synctex=1"
```

3. A REFRESHER COURSE; NO ACTION REQUIRED

TeXShop creates a folder `~/Library/TeXShop` containing several subfolders. One of these folders is named “Templates”; it contains templates for various kinds of TeX documents. I’ll use that folder as an example, although it has not changed recently. Users can edit these templates, add templates of their own, and throw away inappropriate templates. TeXShop displays these templates in a pulldown menu on the source toolbar; users select a template to insert its source code in their document.

When TeXShop is upgraded, there might be new versions of the default templates, but it would certainly be inappropriate to reach into the user’s carefully edited Templates folder and change its contents. Therefore, TeXShop upgrades install the new templates in `~/TeXShop/New`. Users can examine them at their leisure and activate ones they like by moving those to `~/TeXShop/Templates`.

For the record, there is a way to obtain the exact contents of `~/Library/Templates` as seen by a brand new user. To get it, quit TeXShop and move the entire Templates folder to the desktop. Then restart TeXShop. When it discovers that the Templates folder is completely missing, TeXShop replaces it with a new default copy. The same mechanism works with any subfolder of `~/Library/TeXShop`. However, this drastic action should almost never be necessary because of changes introduced in TeXShop 2.33.

4. A SLIGHT UPGRADE CHANGE FROM 2.33 ONWARD

In the past, upgrades did not modify any folder in `~/Library/TeXShop`. Starting with TeXShop 2.33, three folders are touched: `bin`, `Engines/Inactive`, and `Scripts`. Extra files added to those folders by users are left unchanged, but default TeXShop files are replaced by new versions and new default TeXShop files are added. Note that active Engine files are not changed because they don’t live in the Inactives folder.

These changes make upgrading engines much easier. Support files are automatically upgraded, so users only need to upgrade by hand the actual engine files, which seldom change.

5. EASY STEPS FOR SOME USERS

Some TeXShop users have never edited files in `~/Library/TeXShop`, except perhaps to modify the default Templates, have never added new Macros to the Macro menu, have never added new Engines, and have never used Command Completion or at least never added words to the Completion Dictionary. These users can complete the upgrade easily. Quit TeXShop, open `~/Library/TeXShop`, and move the following three folders to the desktop: `CommandCompletion`, `Macros`, `Keyboard`. Then restart TeXShop. Done. But it might be nice to keep the old desktop copies of these folders for a few days in case you made a modification you had forgotten, and find that TeXShop’s behavior has changed.

6. NEW AND IMPROVED ENGINES

In version 2.30, Nicola Vitacolonna made beautiful new engines for metapost and metafun. To obtain these engines, move `nv-metafun.engine` and `nv-metapost.engine` from

`~/Library/TeXShop/Engines/Inactive/Metapost`

to `~/Library/TeXShop/Engines`.

The XeTeX and XeLaTeX engines have been modified to contain the `file-line-error` flag. To obtain these new versions, move `XeTeX.engine` and `XeLaTeX.engine` from

`~/Library/TeXShop/Engines/Inactive/XeTeX`

to `~/Library/TeXShop/Engines`, replacing the older versions there now.

In 2.31 there is a new Sage engine by Dan Drake. If you used the old Sage engine, you will need to switch to the new one because SageTeX is now included in Sage. Because there are several changes in SageTeX, you need to read “About Sage” in

`~/Library/TeXShop/Engines/Inactive/Sage`

for important details.

The latexmk engines maintained by Herbert Schulz are upgraded regularly. Schulz has modified these engines and their support files so that in the future upgrades will occur automatically without user action. But users must make one change in 2.32 to switch to the new files; if this change is not made, the old latexmk from TeXShop 2.30 will stay in place and continue working.

Changing is easy for most users. Latexmk comes with six engine files, located in

`~/Library/TeXShop/Engines/Inactive/Latexmk`

Drag new versions of those which you use from this location to `~/Library/TeXShop/Engines`, replacing the older versions there now. Done.

A small number of users may have edited support files for latexmk that used to be in `~/Library/TeXShop/bin`. These edited versions will remain unchanged in this location, but new support files are now provided in `~/Library/TeXShop/bin/tslatexmk`. In the new latexmk, these support files should not be edited because they will be upgraded by TeXShop upgrades. Instead, Schulz has provided a mechanism to add personal changes to a new editable file. Read the documentation in

`~/Library/TeXShop/Engines/Inactive/Latexmk`

to see how this is done. It is only necessary to take action if you edited the previous support files.

7. MACROS; NO NEW CHANGES AFTER 2.30

Alan Munn provided a wonderful new macro named “Paste Spreadsheet Cells.” Using his macro, you can copy cells from a spreadsheet and paste these cells, embedded in appropriate TeX code, into your source. To obtain the macro, go to

```
~/Library/TeXShop/New/Macros
```

and copy the file `PasteSpreadsheetCells.plist` to the desktop. Then open TeXShop and in the Macro menu select “Open Macro Editor.” Select the “Add macros from file...” item in this menu, navigate to the desktop copy of `PasteSpreadsheetCells.plist`, and choose it. A new “Paste Spreadsheet Cells” macro will be added to your Macro list. If you desire, drag it to a different spot in the list, and then hit the Save button.

8. KEYBOARD SHORTCUTS; NO NEW CHANGES AFTER 2.31

TeXShop has the ability to remap Keyboard Shortcuts. This feature was activated by only a few users, and stopped working some time ago because the file controlling it contained comments within comments, which is illegal in xml. If you modified Keyboard Shortcuts in the past, copy the file

```
~/Library/TeXShop/Menus/KeyEquivalents.plist
```

to the desktop. Then in all cases find the file

```
~/Library/TeXShop/New/Menus/KeyEquivalents.plist
```

and move it to the folder

```
~/Library/TeXShop/Menus
```

overwriting the old file. The new file is only an template explaining how to make changes, but the changes it makes are commented out. In the unlikely event that you edited the old `KeyEquivalents.plist`, merge in your changes from the desktop copy.

9. COMMAND COMPLETION; NO NEW CHANGES AFTER 2.31

TeXShop has Command Completion. Type the beginning of a command and hit the Escape key. TeXShop will complete the command. If several completions are possible, hit Escape several times to cycle between them. The list of known completions is stored in `~/Library/TeXShop/CommandCompletion` and can be edited within TeXShop.

This facility has been expanded by Herbert Schulz in version 2.30. To use his additions, you need a new `CommandCompletion` file. If you modified the default `Command Completions` sometime in the past, copy the file

```
~/Library/TeXShop/CommandCompletion/CommandCompletion.txt
```

to the desktop. Then in all cases find the file

```
~/Library/TeXShop/New/CommandCompletion/CommandCompletion.txt
```

and move it to the folder

```
~/Library/TeXShop/CommandCompletion
```

overwriting the old file. If you modified the old file, you must edit CommandCompletions.txt with TeXShop or TextEdit and merge in your changes from the desktop copy.

10. DOCUMENTATION; NO NEW CHANGES AFTER 2.30

The “Paste Spreadsheet Cells” macro by Alan Munn is documented in TeXShop Help under Macros Help, Default Applescript Macros.

Herbert Schulz’s extensions to Command Completion are explained in a short paper he wrote, which can be found in ~/Library/TeXShop/New/CommandCompletion.

Nicola Vitacolonna’s new engines for MetaPost, nv-metafun and nv-metapost, are explained in his ReadMe in ~/Library/TeXShop/Engines/Inactive/MetaPost. This folder also contains a folder of examples.