ABOUT TEXSHOP 3.50

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.

2. Changes in 3.40, 3.41 through 3.50

Version 3.40 was never released. Versions 3.41 through 3.50 require no user actions.

3. Changes in 3.39

The most significant change in 3.39 is the addition of Michael Sharpe's "Recent TeX Fonts" document and associated font Macros. Michael and I attended the TeX User Group meeting in Portland, Oregon at the end of July, 2014. I knew him as an Applescript expert; several of his scripts are in TeXShop. Other macros from him will appear in a future version. To see his scripts, go to https://dl.dropboxusercontent.com/u/3825336/TeX/index.html. Pay particular attention to *macrocopier.zip* on this location, a stand alone program which makes it easy to maintain and extend TeXShop macros.

At the TUG meeting, I discovered that Sharpe is a font expert widely known to users on other platforms. TeX Live contains a very large number of TEX fonts, but it is not that easy to use them. Most font sets don't have mathematical symbols, and it becomes a design task to find pleasing combinations of fonts for text, sans-serif sections, and mathematics.

Sharpe wrote a document called *Recent TeX Fonts*, now available in the TeXShop Help menu. This document describes a number of pleasing font combinations, one per page. Each page lists the features of a set, provides an extensive sample of text and mathematics

typeset using it, and contains the exact LaTeX code needed to use the font set. These sets are the result of extensive work by Sharpe; I understand that some of them took four months to perfect.

One way to use the document is to select an article or book written with standard fonts and copy Sharpe's implementation section into the document's header and retypeset.

To make this even easier, Sharpe slightly modified the LaTeX template which comes with TeXShop, defining a section in the header bounded by *%SetFonts* comments. The space between two such comments can be empty when the document is originally written. Sharpe defined three macros called "GetFontSet," "SaveFontSet," and "TestFontSet." The first of these brings up a small dialog listing known font sets. When one is selected, its implementation code is written to the source document between the *%SetFonts* comments, replacing other implementation code there. So with one click and one typeset, the document can be seen written with a new set of fonts.

Users can also put their own implementation code between the *%SetFonts* comments. The SaveFontSet macro reaches between the comments and saves the implementation code to a file in `~/Library/TeXShop/bin` named *SetFonts*, which is used by the GetFontSet macro to list known font combinations. Thus *SetFonts* gradually builds into a library of known font combinations.

I'll let the TestFontSet macro speak for itself.

To use these Sharpe additions, it is necessary to use the full TeX Live as installed by MacTeX, because BasicTeX doesn't contain many fonts. All the font sets defined by Sharpe have been tested and are available with TeX Live 2014, except two. The garamondx font has a license permitting free personal use provided the font is not sold. This font is on CTAN, but it cannot be in TeX Live because TUG sells a DVD containing TeX Live. However, a script named getnonfreefonts is available to download and install this font. See `https://www.tug.org/fonts/getnonfreefonts/`.

The *SetFonts* template also has a Lucida entry. Lucida is a commercial font, sold by TUG and others. See `https://www.tug.org/store/lucida/index.html`. Many users own this set, and Sharpe's detailed and non-trivial code to use them will help them obtain the most from the fonts.

It is our hope that the existence of these easy techniques will lead to more TeX documents that don't scream out "I was written with TeX," and instead have a professional printed look.

To use these new features, some customization of TeXShop is required. (The customization isn't necessary if you are running TeXShop for the first time.) To perform the steps, use the Finder's "Go" menu to go to ~/Library/TeXShop, a folder which contains many subfolders.

- The TeXShop folder contains a subfolder named New, which has a subfolder named Templates. This folder contains one template: LaTeXTemplate.tex. Copy it to the active ∼/Library/TeXShop/Templates folder. The LaTeXTemplate.tex folder will replace an existing template. If, of course, you edited this template, then you won't want to overwrite it and instead you'll need to merge the *%SetFonts* comments from the new LaTeXTemplate.tx into your customized template.
- Inside New there is another folder named Macros/FontSets. This folder contains five items. The "refreshfront" item can be ignored, since a copy has already been placed in ∼/Library/TeXShop/bin. The "bg8.pdf" file needs to be placed in ∼/Library/texmf/tex/latex, and you may have to create these folders inside ∼/Library. Finally, the three files GetFontset.plist, SaveFontSet.plist, and TestFontSet.plist contain the new macros, which need to be loaded into the Macros menu.
- If you have never modified the Macros which come from TeXShop, then quit TeXShop and move the entire Macros folder at ∼/Library/TeXShop/Macros to the trash. When you restart TeXShop, a new Macros folder will be created containing the old macros and the three new macros, and you are done.

  But if you edited the macros, then run TeXShop and select Open Macro Editor in the Macros menu. In the TeXShop Macros menu, select the item *Add macros from file.* Navigate to one of GetFontset.plist, SaveFontSet.plist, or TestFontSet.plist and select it. A new macro will appear. Drag it to an appropriate spot, perhaps just under "Open quickly..." Repeat for the other two macro files, and then save the changes.

4. Changes in 3.36, 3.36.1, 3.36.2, 3.37, and 3.38

No user actions are required for these versions.

5. Changes in 3.35

TeXShop 3.35 is compiled on Mavericks. Due to bugs in Lion, it was necessary to disable magnification in the Preview window and selection of rectangles in the Preview window on that system. If at all possible, Lion users should upgrade to Mountain Lion or Mavericks, where these features are restored. Lion users who cannot upgrade might consider using version 3.26 of TeXShop, permanently available on the TeXShop web page. Magnification and rectangle selection work on that version (but the necessary code does not compile on Mavericks). Version 3.26 will not be upgraded.

TeXShop 3.35 contains a GotoLabel macro by Micael Sharpe. New users will obtain it automatically, but older users need to go to ˜/Library/TeXShop/New/Macros and find

the macro in a folder named GotoLabel, which also has a README explaining how to install it.

6. Changes in 3.32, 3.33, 3.34

Versions 3.27 - 3.31 were never released. For version 3.32, TeXShop source code was converted to ARC and compiled with Automatic Reference Counting turned on. Thus memory management is done automatically by Apple's compiler.

Version 3.34 has new LilyPond and MetaPost engines by Nicola Vitacolonna. Active versions of the metapost engines are included in TeXShop. To update the active engines, go to ~/Library/TeXShop/Engines and replace the current engines named "nv-metafun.engine" and "nv-metapost.engine" by the new versions of these engines in

~/Library/TeXShop/Engines/Inactive/MetapostEngines/Engines.

7. Changes in 3.26

Four Applescript macros have problems with recent versions of OS X: Open Quickly, Insert Reference, New Array, and New Tabular. These were fixed by Michael Sharpe. If you use default Applescript Macros and haven't edited them or added your own Macros, you can obtain the new versions by quitting TeXshop, going to ~/Library/TeXShop and moving the Macros folder to the desktop, and then restarting TeXShop. A new version of the Macros file will be created. If all is well, the old Macros folder can be thrown away.

Otherwise, find the files InsertReference-OpenQuickly.plist and NewTabular-NewArray.plist in ~/Library/TeXShop/New. In the Macros menu select "Open Macro Editor". Then select "Add macros from file..." and add the macros from both of these files. The new macros will appear at the bottom of the Macro List. Use the Macro Editor to replace the old contents of the four macros with the new contents using Copy and Paste. Finally throw the new copies at the end of the Macro list away.

8. Changes in 3.25

The TeXShop web site is http://pages.uoregon.edu/koch/texshop/texshop.html, but used to be http://www.uoregon.edu/~koch/texshop/texshop.html. This is changed throughout TeXShop. Users need to edit one of the TeXShop macros to make the change there. Open the Macros editor and edit the "Applescript/TeXShop home page" macro to point to the modern link.

9. Changes in 3.24

No user actions are required for this version. Just one bug is fixed. The Sparkle upgrade mechanism failed on Mavericks for many version up to and including 3.23. This is fixed.

10. Changes in 3.22 and 3.23

TeXShop 3.23 fixed one bug. In 3.22, if the user configured TeXShop to use an external editor and then set the hidden preference ExternalEditorTypesetAtStart to YES, the program crashed when opening a file. This is fixed in 3.23.

No user actions are required for version 3.22 unless you use the Sage engine. If you use Sage, go to ∼/Library/TeXShop/Engines/Inactive/Sage, read "About Sage", make the required changes, and switch to the new Engine file.

11. Changes in 3.21

Versions 3.19 and 3.20 were never released. No user actions are required for version 3.21.

Users with Portables connected to large monitors may want to make one change. For a long time we have recommended that users new to TeXShop arrange the location and size of the Source and Preview windows on the desktop as desired. Most users prefer a side by side configuration with the Source window on the left and the Preview window on the right. Then activate the source window and in the Source menu select "Save Source Position". Similarly activate the preview window and in the Preview menu select "Save Preview Position". From that point on, all TeXShop source and preview windows will appear in the selected positions.

If you have a portable connected to a large monitor, this configuration works as long as you are attached to the monitor. But when you are traveling, the windows will appear on the screen of your portable, and probably not in ideal positions. To fix this, arrange a source and preview window on your portable screen; the portable can be attached to the large monitor at the time. Then activate the source window and in the Source menu select "Save Source Position for Portable". Activate the preview window and in the Preview menu select "Save Preview Position for Portable". After this step, windows will appear in the desired position when you are connected to the large monitor at home or office, and windows will appear in the desired position on the portable screen when you are traveling.

12. Changes in 3.18

User action is only required if SyncTeX is incorrectly configured. Users should check that

- in TeXShop Preferences under the Typesetting tab, the "Sync Method" is set to SyncTeX;
- in TeXShop Preferences under the Engine tab, the two configuration lines for "pdfTeX" each contain the following flags

  ```
  --file-line-error --syncTeX=1
  ```

- in TeXShop Preferences on the same page, the two "TeX + dvips + distiller" lines contain the following instruction

  ```
  --extratexopts "-file-line-error -synctex=1"
  ```

  The easy way to do this is to push the four "Default" buttons beside these four entries.

13. Changes in 3.17

The latexmk engines have been modified. By default, ∼/Library/TeXShop/Engines contains two such engines, pdflatexmk.engine and sepdflatexmk.engine. Drag the new versions of these engines, which are in ∼/Library/TeXShop/Engines/Inactive/Latexmk, into this location.

14. Changes in 3.16

No changes require user action.

15. Changes in 3.15

No changes require user action *unless* you use Monoco as an editing font. Apple has optimized the text and font rendering in Mountain Lion for the Retina display. On a non-Retina display running on Mountain Lion, some fonts in small sizes will appear slightly blurry. To switch back to the old behavior, issue the following command in Terminal:

```
defaults write TeXShop NSFontDefaultScreenFontSubstitutionEnabled -bool YES
```

16. Changes in 3.12 - 3.14

Versions 3.12 and 3.13 were never released. If you have already made changes for 3.11, the only change required for 3.14 is to move the files *nv-metafun.engine* and *nv-metapost.engine* from ∼/Library/TeXShop/New/Engines to ∼/Library/TeXShop/Engines, replacing old versions of these engines. The new metapost engines are by Nicola Vitacolonna. Documentation about them is in ∼/Library/TeXShop/Engines/Inactive/MetaPostEngines-1.4.4.

17. Changes in 3.11

The ConTeXt engines have been renamed. This is the only change. I promised to make this change a year ago, but checking MacTeX-2012 shortly before release, I found that the promise was ignored. The old and new names are

- ConTeXt-MKIV.engine → ConTeXt (LuaTeX).engine
- ConTeXt-xetex.engine → ConTeXt (XeTeX).engine
- ConTeXt .engine → ConTeXt (pdfTeX).engine

The new names make explicit the TeX program which will run the ConTeXt macros for that engine.

The new files were placed in ∼/Library/TeXShop/Engines/Inactive folder. To complete the change, remove the old files from ∼/Library/TeXShop/Engines/Inactive/ConTeXt, and rename ConTeXt-MKiv.engine to ConTeXt (LuaTeX).engine in ∼/Library/TeXShop/Engines.

18. Changes in 3.10

Two important changes in TeXShop 3.10 require user intervention. You may want to read this entire section before making the changes, but here are instructions when you are ready:

- Go to TeXShop Preferences and select the Engine tab. Press both "Default" buttons in the pdfTeX section of the pane in the middle of the left side.
- Go to the Library folder in your home directory. This folder is hidden in Lion, but you can get to it from the Finder's "Go" menu by pressing the option key while the menu is selected. In this Library folder, there is a TeXShop folder containing several subfolders. Inside TeXShop/New/Templates is a template named "Latex-Template.tex". Move this to TeXShop/Templates, overwriting the old template with the same name. If you edited this old template, you'll need to merge your changes with the changes from the new version.

The key feature of TeXShop 3.10 is removal of the "–shell-escape" flag for the pdftex and pdflatex engines. This flag gives TEX permission to call other programs during typesetting. Unfortunately, the flag allows *any* shell program to be run, including for instance a short command which erases everything in your home folder. So when the flag is set, you could download a TEX file from a malicious web site, typeset it, and lose all of your files.

Pushing the two "Default" buttons in Preferences removes the flags.

Why was this flag activated in previous versions of TeXShop? A primary reason is that many users have old documents containing eps illustrations. While pdfTeX and pdfLaTeX can accept illustrations in many formats, including pdf, jpg, and png, they cannot accept eps illustrations. But the *epstopdf* package can call Ghostscript during typesetting and automatically convert eps illustrations to pdf illustrations. The flag gave pdfTeX permission to call Ghostscript.

Two years ago, the TeX Live distribution made this easier and safer. A new "restricted shell escape" command was added to TEX giving TEX permission to call a carefully limited list of programs during typesetting, including Ghostscript. Then the *graphicx* package was modified to automatically convert eps illustrations to pdf format during typesetting, without explicitly including epstopdf. This means that old projects with eps illustrations will automatically typeset with pdflatex, provided the source includes the *graphix* package.

At the time of these changes, we did not remove "–shell-escape" because it was still needed to convert tiff files to png format automatically during typesetting. This conversion is done by */usr/bin/convert* , a program from ImageMagick which is installed as part of MacTeX. Unfortunately, TeX Live could not add convert to the restricted shell escape list because in the Windows world there is a program with this name which can do dangerous things. Also we knew that tex4ht, which can be used to typeset TEX source and output html pages, calls convert.

Recently, we discovered that while the tex4ht *script* calls convert, the convert program is not called by the pdflatex portion of the script. So the flag is not needed for tex4ht.

TeXShop 3.10 contains a new command in the File menu called "Convert tiff." This item is active when a TeX source window is active. Selecting the item brings up a dialog listing all files and folders in the directory containing the source file. Only tiff files are active in this dialog; others are grayed out. Multiple items can be chosen in the dialog. Pushing the "Convert" button at the bottom of the dialog will create png versions of all selected tif or tiff illustrations.

The *Convert tiff* dialog could be strengthened in the future to provide other graphic conversions, if there is sufficient demand.

The new Latex Template removes the inclusion of epstopdf because it is no longer necessary, and removes code to automatically convert tiff files to png using *convert* because it

will not run without "–shell-escape" and has been replaced with the new TeXShop menu command.

There is one final change in TeXShop 3.10. For many years, TeXShop has provided two different ways to create projects governed by a master root file with several include files. The earlier method used a menu command *Create Project Root*. The later more powerful method involves adding "% !TEX root = ..." to the first few lines of included files. This is the preferred method.

We have removed the menu item *Create Project Root*. Old projects using this mechanism will still typeset correctly, but new projects should use the "% !TEX root = ..." syntax instead.

19. Changes in 3.07, 3.08, and 3.09

The pdflatexmk engine is now active by default for new users. Older users can activate it by finding "pdflatexmk.engine" in ∼/Library/TeXShop/Engines/Inactive/Latexmk and drag it to ∼/Library/TeXShop/Engines. No other 3.07, 3.08, or 3.09 changes require user intervention.

20. Changes in 3.05 and 3.06

In 3.05, the folder ∼/Library/TeXShop/CommandCompletion has a new subfolder named GratzerMathCC. This folder contains an optional CommandCompletion.txt file with additional completions by George Gratzer, useful when typesetting mathematics. It also contains documentation by Gratzer and Herbert Shultz about these extensions.

To obtain the folder, move it from ∼/Library/TeXShop/New/CommandCompletion to ∼/Library/TeXShop/CommandCompletion. To activate the new completions, move CommandCompletion.txt from the new folder up a level to the main CommandCompletion folder, replacing the similarly named file there now.

No additional changes were made in 3.06.

21. Changes in 3.00 - 3.04

Version 3 requires Lion. It is the start of a new sequence of TeXShop Lion releases.

For some time to come, the old 2.** series of TeXShop releases for Tiger, Leopard, and Snow Leopard and the new 3.** releases for Lion will be developed in parallel. The Sparkle update system will only upgrade TeXShop 2.** to later 2.** releases and will only upgrade TeXShop 3.** to later 3.** releases. It will never upgrade, say, TeXShop 2.43 to TeXShop 3.02.

Versions 3.00 and 3.01 were renamed TeXShop-64. This broke several applescript macros, and turns out to be unnecessary for separating the 2.** and 3.** releases in Sparkle. So version 3.02 has the old TeXShop name.

For a detailed list of Lion features, see the Lion section of the TeXShop Help Panel, available under the TeXShop Help menu. The Internationalized versions of this Panel are often out of date. Changes are also described at http://pages.uoregon.edu/koch/texshop under the Lion tab.

22. Changes in 2.42 and 2.43

There are revised engines in ∼/Library/TeXShop/Engines/Inactive for ConTeXt running on top of LuaTeX, XeTeX, and pdfTeX. In particular, the ConTeXt (LuaTeX) engine is now a default engine, replacing ConTeXt-MKIV in previous versions of TeXShop. Drag this file from ∼/Library/TeXShop/Engines/Inactive/ConTeXt to ∼/Library/TeXShop/Engines.

23. Changes in 2.41

The only change is an updated Command Completion.pdf document. Find this document in

∼/Library/TeXShop/New/CommandCompletion

and move it to

∼/Library/TeXShop/CommandCompletion

24. Changes in 2.40

If you already have 2.39, no changes are needed for 2.40.

25. Changes in 2.39

One new item is provided for ∼/Library/TeXShop/CommandCompletion: a French translation of Herbert Schulz's documentation named Completement2011.pdf. Find this item in

∼/Library/TeXShop/New/CommandCompletion

and move it to

∼/Library/TeXShop/CommandCompletion

26. Changes in 2.38

Three new items are provided for ∼/Library/TeXShop/CommandCompletion: a revised document *Command Completion for TeXShop.pdf* and two folders named IndentedCC and Quick Start Guide for Command Completion. Find these three items in

∼/Library/TeXShop/New/CommandCompletion

and move them to

∼/Library/TeXShop/CommandCompletion

The documentation in ∼/Library/TeXShop/Engines/Inactive has been revised, but the revised version is installed automatically, so no action is needed when upgrading. According to the revised documentation, users of the ConTeXT-MKIV.engine must run the following command ONCE in Terminal before the engine will work:

```
luatools --generate
```

27. Changes in 2.35, 2.36, and 2.37

If you already have 2.36, no changes are needed for 2.37.

If you already have 2.34 and are upgrading to 2.37, read this section and ignore everything else. The only significant change is that there is a new active engine: ConTeXt-MKIV. This is the version of ConTeXt running on top of LuaTeX.

Find the ConTeXt-MKIV.engine in ∼/Library/TeXShop/Engines/Inactive/ConTeXt. Drag or copy it to ∼/Library/TeXShop/Engines. Done.

28. Changes in 2.34

If you already have 2.33 and are upgrading to 2.34, read this section and ignore everything else. The only significant change is that there is a new active engine: LuaLaTeX. That is because LuaLaTeX, under development for several years, has reached the stage in TeX Live 2010 when it can be used for serious work. TeX Live 2010 will be released shortly.

Find LuaLaTeX.engine in ∼/Library/TeXShop/Engines/Inactive/LuaTeX. Drag or copy it to ∼/Library/TeXShop/Engines. Done.

29. 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"
```

30. 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.

31. 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.

32. 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.

33. 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.

34. 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.

35. 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.

36. Command Completion Changes In 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.

37. Documentation Changes In 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.