

# Using latexmk with T<sub>E</sub>XShop

Herbert Schulz  
herbs2@mac.com

## 1 What is latexmk?

Compiling a tex file containing cross-references, bibliographic references and/or indexes is a multi-pass process; i.e., you've got to run (pdf/x<sub>e</sub>/lua)latex multiple times with possible intermediate runs of bibtex and/or makeindex until all references are resolved. The latexmk perl program, rewritten and presently maintained by John Collins<sup>1</sup>, automates this multi-pass process. By default it first runs (pdf/x<sub>e</sub>/lua)latex on a source file, determines file dependencies by examining the log, aux, etc. files produced by the run and then automatically runs bibtex<sup>2</sup> and/or makeindex, if needed, and the correct number of additional runs of (pdf/x<sub>e</sub>/lua)latex to generate the bibliography, index and cross-references. Recent versions of latexmk also work correctly with the nomencl package as well as the glossary and glossaries packages and other packages that produce multiple bibliographies or indexes.

## 2 Quick Start!

This section will get you started quickly. Unless you are trying to customize the behavior of the supplied engines or want to use the more esoteric engines this really is all you need!

### 2.1 Quick Install.

To activate the latexmk engine files simply move all the files with extension .engine from ~/Library/TeXShop/Engines/Inactive/Latexmk two folder levels up, to ~/Library/TeXShop/Engines, and (re-)start T<sub>E</sub>XShop. (Note: ~/Library is the Library folder in your HOME folder. You can use the TeXShop → Open ~/Library/TeXShop menu item to open that folder.) When you click on the popup engine menu on the Source toolbar the newly enabled engine names should appear; see Figure (1) to see how that menu changes. **Note: the engine names will *not* appear in the Typeset Menu.**

### 2.2 Quick Use.

At the top of your Source file place the line

```
% !TEX program = pdflatexmk
```

to use the pdflatexmk engine which will use pdflatex to typeset your document. Substitute latexmk, xelatexmk or lualatexmk for pdflatexmk to use latex, xelatex or lualatex to typeset your Source. From then on simply using Typeset → Typeset (Cmd-T) will run through the complete process of fully typesetting your document.

## 3 What is here?

There is a set of ten engine files to be placed in ~/Library/TeXShop/Engines. There is a tslatexmk folder already placed in ~/Library/TeXShop/bin. The files in that folder consist of the latexmk program, ten basic initialization (rc) files used by the ten engine files, a common file for personal settings (latexmkrcDONTedit) and two shell scripts used for pdftricks and pst-pdf figure

---

<sup>1</sup>The latexmk web site is <<http://www.phys.psu.edu/~collins/software/latexmk-jcc/>>. You can get the latest version of latexmk at <<http://www.phys.psu.edu/~collins/software/latexmk-jcc/versions.html>>.

<sup>2</sup>As of version 4.22 latexmk will automatically choose between running bibtex or biber as required.

processing. When any of the new engines is first run the latexmkrcDONTedit file will automatically be copied to `~/Library/TeXShop/bin/latexmkrcedit` if it doesn't already exist. You may copy the file there manually if you wish. **Any changes or additions to the configuration (e.g., new dependencies and rules) should be placed in the latexmkrcedit file. When TeXShop is updated the files in the ~/Library/TeXShop/bin/latexmk may automatically get updated; don't edit them or your changes may get lost.**

## 4 What is New with this Version

### 4.1 Command Line Options

Most of the latest updated engine files now allow the passing of command line options to the underlying typesetting engine using the

```
% !TEX parameter = ...
```

directive in recent versions of TeXShop; e.g., the lines

```
% !TEX program = pdflatexmk
% !TEX parameter = --shell-escape
```

will use pdflatex with shell-escape as the typesetting engine. You can pass more than one option so that

```
% !TEX parameter = --shell-escape --nonstopmode
```

will pass both of those options to the typesetting engine.

Note: the dtxmk, pdfricksmk and pst-pdfmk engines have *not* been updated because of their specialized use.

### 4.2 Project Customization File

The engine files supplied with this version of latexmk for TeXShop now allow you to have a platexmkrc file, containing latexmk configuration information, in the same folder as the file you typeset (i.e., the root file for a distributed document). This can be useful if your project needs special configuration options for a certain task.

E.g., you wish to use texindy instead of makeindex to process the idx file into a ndx file you might include a platexmkrc file in the same directory as the root file of a project with contents

```
$makeindex = "texindy %0 -o %D %S";
```

to use texindy rather than the default makeindex; make sure you end the file with a blank line. You could also add special options to the processing for a particular situation. Make sure you understand the syntax used by latexmk for customizing commands before playing with this feature.

**One warning:** if you are going to use this feature understand that the platexmkrc file will be used for *any* file in that folder that is typeset.

## 5 Using latexmk with TeXShop.

**NOTE: If you are updating to this version of latexmk for TeXShop from a previous version you need only activate the engine files, as noted above, and restart TeXShop after installing the files.**

There are ten engine files; two for running latex (one with a final run through dvips and ps2pdf [latexmk.engine] and one with a final run through dvipdfmx [dvipdfmxmk.engine]), two for using pdflatex [pdflatexmk.engine and sepdfplatexmk.engine] (the second one for use when you need to use --shell-escape: but see Sub-Section (4.1) ), one for using xelatex [xelatexmk.engine], one for using lualatex [lualatexmk.engine], two for using the pdftricks or pst-pdf packages with

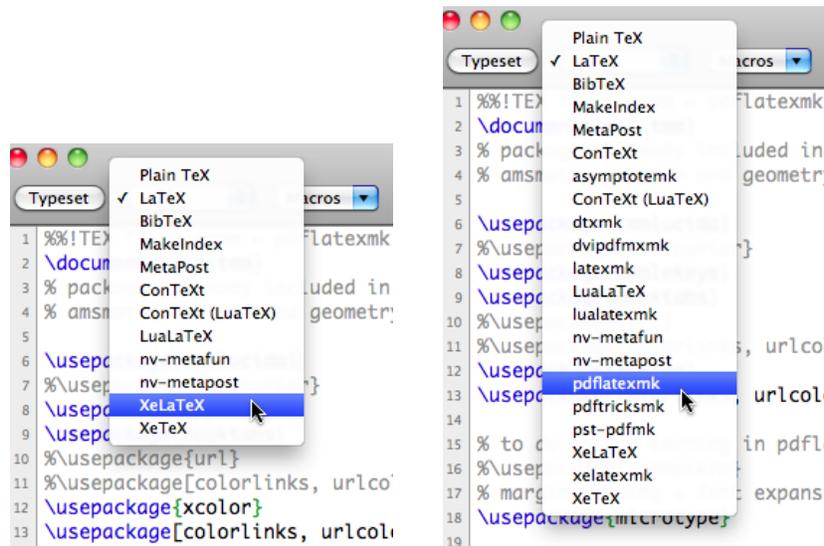


Figure 1: Default and updated versions of the engine pop-up menu after installing the latexmk engines.

pdflatex [pdftricksmk.engine or pst-pdfmk.engine respectively] and one for use with files that use the asymptote package [asymptotemk.engine]. The final engine is a very basic engine for typesetting dtx files for a package into the final documentation [dtxmk.engine]. The exact form of the commands and options used are in the corresponding rc file (e.g., latexmkrc for the latexmk.engine) in `~/Library/TeXShop/bin/tslatexmk` and shouldn't be changed.

You can use these engine files by using the drop down menu on the source tool bar or placing the line

```
% !TEX program = pdflatexmk
```

(for using pdflatex—similar lines for latex and xelatex, etc.) at the top of your document<sup>3</sup>; then simply using Typeset (Cmd-T) will automatically run the proper engine. Note: these engines *don't* appear under the Typeset Menu but only under the pop-up menu on the source toolbar. Figure (1) shows the default and updated pop-up menu after installing the latexmk engine files.

I have only tested these engines with relatively trivial distributed documents (which include other files using `\include` commands) but it appears that latexmk deals with them properly. Note that when compiling a file with name `rootname.tex` a file with name `rootname.fdb_latexmk`<sup>4</sup> is created. This file contains the dependency information for the distributed document so making changes in an included file will force the recompile of the root document by latexmk.

### 5.1 Using the pdftricks package with latexmk.

The pdftricks package allows the inclusion of pstricks graphics in documents compiled with pdflatex. The package generates a file for each postscript figure included in the document. Each of those figure files is then processed to produce a pdf file containing a figure with a tight enclosing bounding box. The pdftricksmk engine included with this version of latexmk processes the original file, the figure files, etc., all only if they have changed. To use the engine place the line

```
% !TEX program = pdftricksmk
```

<sup>3</sup>For the dtxmk engine the line should be placed just below the initial “`\iffalse`” line of the dtx file.

<sup>4</sup>The dependency file had extension `dep` in previous versions of latexmk but didn't do a complete job of keeping track of those dependencies.

at the start of the file and Typeset the file. The processing steps for each of the figure files is latex → dvips → ps2pdf → pdfcrop to ensure the proper bounding box is created for each figure. **NOTE: you must use the [noshell] option to the pdftricks package or latexmk will get into a run-on condition. All figure processing will be taken care of by latexmk.**

## 5.2 Using the pst-pdf package with latexmk.

The pst-pdf package also allows the inclusion of pstricks graphics in documents compiled with pdflatex. When the source file is compiled with latex a dvi file containing all of the figures is created. Further processing through the sequence dvips → ps2pdf → pdfcrop produces a single file that contains all of the figures with proper bounding boxes. A run of pdflatex on the source file then includes all of the figures previously generated. The pst-pdfmk engine takes care of all of the intermediate processing of the figures as well as the final run(s) of pdflatex, etc. To use the engine place the line

```
% !TEX program = pst-pdfmk
```

at the start of the file and Typeset the file.

## 5.3 The glossary, glossaries and such packages.

Packages that produce multiple and custom indexes, glossaries, etc., use one of two naming schemes for the multiple files they create:

1. The first uses standard extensions but special files names for the generated files. Latexmk can keep track of changes in and “knows” how to process these files. The multibib and multind packages are examples that use this method.
2. The second uses the source file name for the file but uses custom extensions to create the files. Latexmk needs “help” to know how to process these files in the form of dependencies and rules. Dependencies tell latexmk what the input and output extensions are and which rule to use to go from input to output. The index, glossary and glossaries packages are examples that use this second method.

In addition, while the glossaries package supersedes the glossary package the order of the file extensions created by acronym and custom lists, processed by makeindex and then read in by subsequent runs of (xe/pdf/lua)latex are reversed in the two packages. This latest version of latexmk configured for T<sub>E</sub>XShop works correctly for both packages. If you need to create your own custom lists see the examples in the latexmkrcedit file for creating dependancies and rules for latexmk.

## 6 What these engines won't do, etc.

There are many features of latexmk that aren't used in these simple engine files. See the documentation for latexmk in the supplied full distribution.

The placement of the latexmk program in ~/Library/TeXShop/bin/tslatexmk/ is non-standard; that directory is not on the standard path. It is possible to put the program in /usr/local/bin/ or use the version of latexmk that is part of MacT<sub>E</sub>X-2008 and later and it will then be usable from the command line. If you use the program in one of those locations you should modify the engine files to reflect the change in location.

Finally, changes in eps files *included in figures* created by the pdftricks or pst-pdf packages are *not* detected by this packaging of latexmk at this time. I hope to correct that problem at a later date.

Try it... I hope you like it.

Good Luck,  
Herb Schulz  
(herbs2@mac.com)