

T_EX Live Utility: A slightly-shiny Mac interface for T_EX Live Manager (tlmgr)

Adam R. Maxwell

Abstract

T_EX Live Utility is a Mac OS X graphical user interface for the T_EX Live Manager command-line tool, `tlmgr`. I'll discuss the goals of the program, several usage examples, and some of the tricky issues in wrapping a tool in order to make it accessible to graphical user interface users. Many of these issues (e.g., error handling and selection of features to expose) are not platform-specific, so could also be of interest to non-Mac users.

1 Introduction

The T_EX Live Manager was introduced in T_EX Live 2008 [4]. It allows users to manage a T_EX Live installation, giving easy access to updates from CTAN (the Comprehensive T_EX Archive Network) for various packages, and allowing global configuration of options such as paper size [2]. It has a graphical user interface (GUI) mode, based on the cross-platform Perl/Tk toolkit, as well as a command-line tool, `tlmgr`.

Although Mac users usually prefer a GUI over a command-line program, the initial response to the Perl/Tk interface ranged from tepid to antagonistic. In general, users felt that it was not “Mac-like”, and that it exposed more features than users commonly needed. Following a discussion on the MacT_EX mailing list, initiated by Jérôme Laurens in October 2008, providing a native GUI for Mac OS X seemed useful.

The benefit of a native GUI is that it can be instantly familiar to a user of a particular platform, and is less likely to require installation of 3rd party libraries such as Perl/Tk in order to work. Ideally, this would serve to reduce complaints and pleas for help from Mac users to the T_EX Live team, thereby insulating them from a vocal and occasionally obnoxious subgroup.¹

I started work on a trivial program on 6 December 2008, and had an alpha release ready for the MacT_EX working group the next day. Whatever your opinion of Apple, their Cocoa frameworks allow rapid development of software by doing much of the work for the developer. After several iterations

¹ My fellow Mac users may resent this generalization. However, it is not uncommon to receive reports from users saying “You need 25 pixels on the bottom of this window, and you have 23. Also, your text field in the preferences settings is 1 pixel misaligned at the right edge. See page thus-and-such of Apple’s *Human Interface Guide*.” I admit that I have sent such reports to Apple, regarding their own software.

and feedback on feature requirements, a beta version was announced on 30 December to the “T_EX on Mac OS X mailing list.”² It was released under the new BSD license, and initially hosted on Google Code at <http://mactlmgr.googlecode.com>. As Google no longer allows hosting of binary downloads, as of August 2014 the project is now maintained on GitHub at <http://github.com/amaxwell/tlutility>.

2 Design

The initial list of features was fairly minimal, and included:

- List/install updates
- List installed packages
- Reinstall `tlmgr` itself
- Change paper size

Over time, various other features have been added, but the most frequently used is the update feature, as far as I know. T_EX users on Mac OS X tend to be compulsive updaters and early adopters, for better or for worse, often updating once a week or even more frequently. To the T_EX Live team’s credit, they have done an excellent job at providing a reliable infrastructure for this purpose.

The overall design goals for T_EX Live Utility have always been to:

1. Expose only the most common tasks
2. Give users a consistent (Mac-like) interface
3. Use `tlmgr`, instead of reimplementing it
4. Give useful feedback for errors
5. Do not require command-line interaction
6. Avoid blocking the GUI with long-running operations

The architecture of T_EX Live Utility has remained largely consistent with the original releases. I could claim this is due to a great design, but it’s also due to a certain amount of inertia on my part; in other words, I’m too lazy to redesign it. Apple recommends a Model-View-Controller architecture ([1] gives a brief introduction), with logic for these general tasks split up into separate objects. It uses the Cocoa frameworks, and is written in Objective-C (and plain C, as Objective-C is a superset of the C language). Since MacT_EX³ requires administrative privileges to install, T_EX Live Utility also requires administrative privileges to update and change various options. This is handled via a separate command-line tool called `tlu_ipctask`, which executes `tlmgr` with root privileges and passes its standard output and

² <https://email.esm.psu.edu/mailman/listinfo/macosex-tex>

³ <http://www.tug.org/mactex>

Package	Status	Installed	Remote	Size
tcolorbox	Update available	34725 (3.11)	34818 (3.12)	3.1 MB
texlive-scripts	Updating...	34740	34889	84.8 KB
xetexko	Update available	34321 (2.7)	34864 (2.8)	256.8 KB
yathesis	Update available	34780 (0.99j)	34842 (0.99j)	1.3 MB
assoccnt	Not installed		34934	221.1 KB
breqn	Not installed		34867	1.1 MB
chemformula	Not installed		34898	951.3 KB
chemgreek	Not installed		34896	539.7 KB
clrscode3e	Not installed		34887	5.9 KB
ghsystem	Not installed		34925	2.4 MB
komacv	Not installed		34906	461.6 KB
luatodonotes	Not installed		34908	219.3 KB
mathtools	Not installed		34868	1.0 MB
mugsthesi	Not installed		34878	359.4 KB
collection-latexextra	Update available	34735	34934	4.3 KB
collection-latexrecomm.	Update available	33855	34868	602.0 B
collection-luatex	Update available	33467	34908	546.0 B
collection-publishers	Update available	34737	34878	962.0 B

Figure 1: The TeX Live Utility main window, with packages listed for update. Note that the uppermost package will be automatically removed, as it has been removed on the server.

standard error back to TeX Live Utility over a Distributed Objects communications channel.⁴

The first and last items of the design goals have driven most of the decisions. Each `tlmgr` command that is invoked by TeX Live Utility is encapsulated in an `NSOperation` subclass. An `NSOperation` is an object which can be enqueued for asynchronous execution on a separate thread. This avoids blocking the GUI thread (“main thread”, in Cocoa parlance) while waiting for a `tlmgr` command to finish executing. Very little data is shared between threads, in order to avoid locking and reduce complexity.

3 Usage examples

The main window of TeX Live Utility is tabbed, and the present iteration of the GUI is intended to give a web browser-style interface to the repository.

3.1 Updates

Figure 1 shows the main window and packages listed for update; this tab is an interface for `tlmgr update` `--list` and allows you to run `tlmgr update foo0 foo1...fooN` for specific packages, or `tlmgr update --all` to update all packages.

3.2 Packages

Figure 2 shows the second tab of the main window, which lists all available packages; this tab is an interface for `tlmgr list`. When a network connection is not available, TeX Live Utility runs `tlmgr list --only-installed`. Actions available from this tab

⁴ <https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/DistObjects/DistObjects.html>

Package	Description	Status
bguaq	Improved quantifier stroke for Begriffsschrift...	Installed
bhcxam	An exam class designed for Mathematics Teachers...	Installed
bib-fr	French translation of classical BibTeX styles	Installed
bibarts	"Arts"-style bibliographical information.	Installed
▼ biber	A BibTeX replacement for users of biblatex.	Mixed
amd64-freebsd	amd64-freebsd files of biber	Not installed
i386-cygwin	i386-cygwin files of biber	Not installed
i386-freebsd	i386-freebsd files of biber	Not installed
i386-linux	i386-linux files of biber	Not installed
i386-solaris	i386-solaris files of biber	Not installed
universal-da...	universal-darwin files of biber	Installed
win32	win32 files of biber	Not installed
x86_64-darwin	x86_64-darwin files of biber	Installed
x86_64-linux	x86_64-linux files of biber	Not installed
x86_64-solar...	x86_64-solaris files of biber	Not installed
► bibexport	Extract a BibTeX file based on a .aux file.	Mixed
bibhtml	BibTeX support for HTML files.	Installed
biblatex	Bibliographies in LaTeX using BibTeX for sortin...	Installed

Figure 2: The TeX Live Utility main window, with packages listed. Note that binaries for various architectures are shown via a disclosure triangle.

Package	Date
Restore 27,227	Aug 9, 2014, 16:33:54
lua ² atex-math	
Restore 31,389	Aug 2, 2014, 11:23:01
lua ² libs	
Restore 33,861	Aug 2, 2014, 11:23:04
lua ² mplib	
Restore 34,021	Aug 2, 2014, 11:23:07
lua ² otfload	
Restore 34,131	Aug 2, 2014, 11:23:11
Restore 34,726	Aug 9, 2014, 16:49:06
lua ² otfload.universal-darwin	
Restore 30,313	Aug 2, 2014, 11:23:09
lua ² otfload.x86_64-darwin	
Restore 30,313	Aug 2, 2014, 11:23:10

Figure 3: The TeX Live Utility main window, with backups listed.

include install, reinstall, and removal of specific packages. Forcible removal can be accomplished by holding down the option key and choosing the “Remove Selected Packages” menu item.

3.3 Backups

The Backups tab shown in Figure 3 gives a listing of available backups, from `tlmgr restore`. This allows the user to select a previous revision of an updated package...one of the most useful features of TeX Live Manager! The default setting for backups can be configured via an options sheet, and I recommend that users keep at least 1–2 versions of packages. Updates from CTAN can occasionally break packages, as they’re not tested at all.

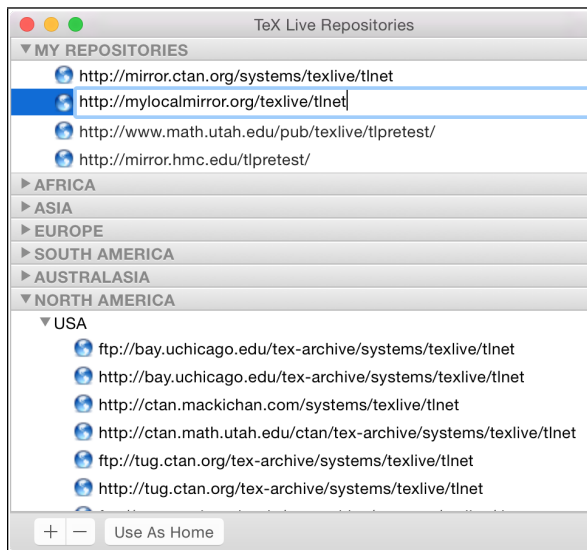


Figure 4: The TeX Live Utility repository interface, showing potential hosts.

3.4 Repositories

Managing and interacting with repositories is one of the more tedious parts of the program. In general, the CTAN multiplexor⁵ works quite well, and is the default repository. It automatically selects a “good” server for your geographic location, and typically provides servers that are up-to-date. However, arbitrary servers often have different sync states with their master CTAN node, so if you run `tlmgr update --list` your next call to `tlmgr update --all` may use a different server and update a different set of packages. For this reason, TeX Live Utility uses the same server for updates as it does for listing them.

Users can also set a default repository, based on the available CTAN mirrors at the time TeX Live Utility was released.⁶ Figure 4 shows the interface for choosing a repository; this is analogous to the “Bookmarks” interface in a web browser, and supports drag-and-drop to/from browsers. Arbitrary repositories can also be added, in the event that you have a private mirror or want to connect to an add-on repository such as `http://tlcontrib.metatex.org`.

Within the TeX Live Utility main window, you can choose a specific mirror by typing part of its name or location (e.g., `usa`) in the location bar, as shown in Figure 5. The location bar also offers standard “Reload” and “Cancel” buttons, and draws a progress bar during package installation (a user’s suggestion, inspired by web browsers such as Safari).

⁵ `http://mirror.ctan.org/`

⁶ This list of mirrors is developed by parsing `http://www.tex.ac.uk/tex-archive/CTAN/sites` using a Python script.

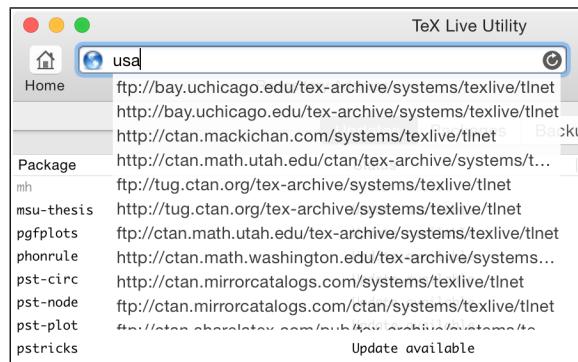


Figure 5: The TeX Live Utility address bar, showing the autocompletion feature.

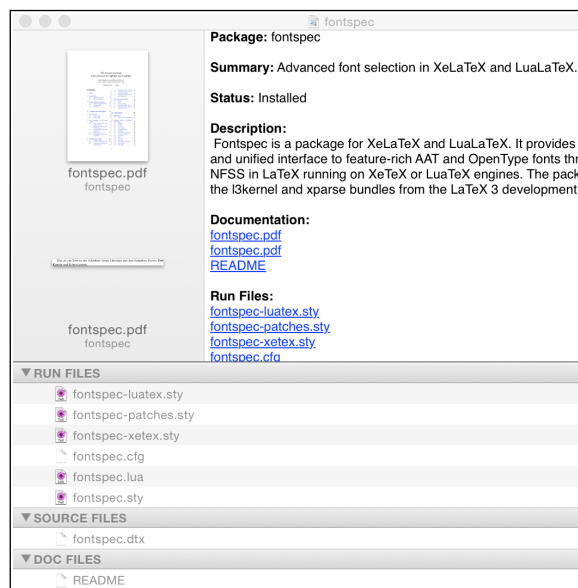


Figure 6: The TeX Live Utility Info panel, showing documentation and various associated files for a given package, along with a short description from the TeX Live Manager database.

3.5 Info

Double-clicking on a given package in one of the main listings will show information for that package, along with the files associated with it in the TeX Live Manager database and any documentation found by `texdoc`. This is shown in Figure 6. For more advanced users, it’s easier just to type `texdoc fontspec` than to launch TeX Live Utility just for this task. However, it’s convenient if you’re browsing the package list and wonder what is (being) installed on your system.

3.6 Logging

I attempted to hide the log window shown in Figure 7 during a UI redesign, but that was not possible. It

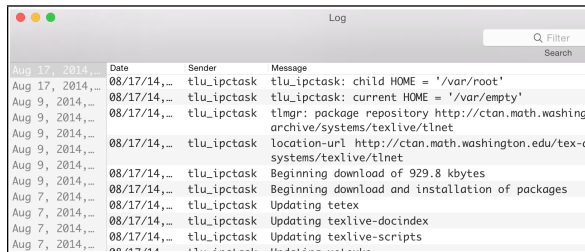


Figure 7: The TeX Live Utility Log window, showing progress and debugging output.

turns out to be a critical feature, given that problems will occur sooner or later during an update, whether due to a problem with the TeX Live infrastructure or a bug in TeX Live Utility itself. Log sessions are saved, and you can select a previous session from the list on the left. Unfortunately, there’s a significant amount of clutter in the list, as TeX Live Utility logs many operations internally; this makes it less generally useful to users, who can easily mistake routine diagnostics for an actual problem.

3.7 Miscellaneous

TeX Live Utility also provides a means to schedule update checking, which ties into the Notification Center on Mac OS X 10.8 and later, and provides an alert dialog on 10.7 and earlier. The update notice lets you start TeX Live Utility and begin the update process, without having to remember to manually launch it and check for updates.

Various preferences can also be set, and there is a nascent feature for installation using the TeX Live network install script. This allows a user to install in an alternate location, and is mainly intended for users who do not have admin privileges. The TeX Live Utility Help menu describes how to enable and use it.

TeX Live Utility attempts to use the OS proxy configuration, and sets the necessary environment variables to work with `tlmgr` and other tools that it uses. This is one of the more complicated features from a code perspective (and one of the more fragile).

Obsolete versions of TeX Live can also be managed. TeX Live Utility will detect whether your installed version is older than the one on the remote repository, and switch to a historical archive as needed.

By holding down the option key, the menu item for reinstalling TeX Live Manager can also install it from the `tlcritical` repository, where test versions of TeX Live Manager are uploaded. This allows Mac users to easily test the latest version, and provide feedback to the TeX Live development team. Re-

verting to the current official version of TeX Live Manager is straightforward, via the same process without the option key.

A Spotlight importer for DVI files is included with TeX Live Utility, as is a Quick Look plugin for DVI files. These aren’t strictly related to TeX Live Utility’s core functionality, but it’s a convenient way to distribute them.

4 Pitfalls

This section contains a mixture of description, complaints, and advice for anyone contemplating a similar exercise, perhaps on another platform. Determining the intent of each piece is left as an exercise for the reader.

4.1 Error handling

Error handling is the biggest challenge in working with `tlmgr`; *viz.*, trying to communicate comprehensible errors to the user. For example, if a call to `updmap-sys` fails as part of the overall update process, there is no way for TeX Live Utility to know which command failed, or give the user any helpful instruction for resolving the problem. The standard error output from `tlmgr` contains the necessary information, but parsing it for individual errors is not a practical solution; each tool such as `fmtutil` or `updmap-sys` has its own error message style, and they are designed to be read by humans rather than parsed by a machine.

4.2 Asynchronous processing

An update can be hundreds of megabytes, especially if you update infrequently. This justifies the need for an asynchronous process model, in my opinion, although that adds complexity. However, running multiple operations at once could be problematic; TeX Live Utility doesn’t allow you to do a restore while doing an update, for instance, although you can show the Info panel for various packages during an update.

4.3 TeX Live Manager updates

The original TeX Live Manager in 2008 had a bug whereby it would delete itself during an update, and at Karl Berry’s suggestion, I modified TeX Live Utility to download and execute the standalone shell script updater from CTAN. This has turned out to be a good thing in the long run, as it also allows easy testing of the `tlcritical` version, and recovery from future installer problems is trivial.

4.4 Environment variables

Environment variables have caused the most painful support issues. Some users alter their `PATH`, `TEX*`, or

PYTHON* environment variables using various obscure and poorly documented techniques.⁷ Unfortunately, they typically forget having done it, so T_EX Live Utility checks for these settings, logs a mildly rude warning message, and unsets their environment variables in its process space. Another clever user set his `umask` to 077 using these techniques, in the name of security. This meant that T_EX Live Utility could not install updates, even as root, and was extremely puzzling.

5 Future plans

Future development plans for T_EX Live Utility are limited at this time, since it works pretty well, and has only a single, very lazy, developer. However, a few things are in the works, presented here in probable order of appearance.

5.1 BasicT_EX documentation

The BasicT_EX distribution provided by Dick Koch is a small subset of T_EX Live, provided for users with limited bandwidth or storage. One of the ways it saves space is by not including documentation, but some users have requested the capability to install documentation for packages on a case-by-case basis.

5.2 Interface updates

A few cosmetic problems exist on Mac OS X 10.9 and later; in fact, it's surprising that users have not yet pointed out that the progress bar should be drawn differently. More importantly, I plan to add toolbar buttons for the update action once again.

5.3 Network install

The network installer (a hidden feature) needs to be refined, and options simplified so that it's more usable; manually editing cryptic shell script variable names, needed at present, does not meet T_EX Live Utility's design goals! It should also integrate with the T_EX Distribution structure on Mac OS X [3].

⁷ This used to be via `~/MacOSX/environment.plist`, and can now be accomplished via other means. I strongly advise against this, as it affects all programs spawned by the `loginwindow`, but is overridden in the Terminal by shell init files. It can literally take days of email exchanges to sort these problems out.

5.4 Privileged code

Apple has deprecated the `AuthorizationExecuteWithPrivileges()` C function that T_EX Live Utility uses to run tasks as root, and is currently recommending that developers use `launchd` to run their privileged process. This will require significant effort, mainly in rewriting the communication code between the two processes.

6 Acknowledgments

This program would not be as effective as it is without the testing and feedback of Bruno Voisin, Herb Schulz, Justin C. Walker, Dick Koch, Will Robertson, and other members of the MacT_EX group. The icon was drawn by Jérôme Laurens, and I thank him for letting me use it for this project.

Thanks to the T_EX Live team for being supportive, especially Karl Berry in suggesting which features to include (or not!). Norbert Preining added the `--machine-readable` option to `tlmgr`, so I could do away with my gruesome ad-hoc parsing code, and has made numerous other improvements as maintainer of T_EX Live Manager. Hopefully T_EX Live Utility makes their lives a bit easier, in that they have to deal with fewer obnoxious Mac users like me!

References

- [1] Pavneet Arora. YAWN — A T_EX-enabled workflow for project estimation. *TUGboat*, 33(2):196–198, 2012. <http://tug.org/TUGboat/tb33-2/tb104arora.pdf>.
- [2] Karl Berry, editor. *The T_EX Live Guide—2014*, 2014. Available online: <http://tug.org/texlive/doc/texlive-en/texlive-en.pdf>.
- [3] Richard Koch. Support for multiple T_EX distributions in i-Installer and MacT_EX. *TUGboat*, 28(3):329–334, 2007. <http://tug.org/TUGboat/tb28-3/tb90koch.pdf>.
- [4] Norbert Preining. T_EX Live 2008 and the T_EX Live Manager. *ArsT_EXnica*, 6:67–75, 2008. <http://www.guitex.org/home/numero-6>.

◇ Adam R. Maxwell
 Port Angeles, WA 98362
 USA
[amaxwell \(at\) mac dot com](mailto:amaxwell@macdot.com)
<http://github.com/amaxwell/tlutility>