

# TUG-GUI-APPLICATIONS

RICHARD KOCH

## 1. GUI APPLICATIONS

This is the portion of the TUG documentation which deals with making the GUI Applications package.

**Warning:** It is important that the folders in which packages are built live on the hard disk rather than an external disk. Building constructs folders named “root” which are copies of material to be installed. This root folder and its contents are often owned by root. But material on an external drive is owned by the user instead.

A clean package file will contain

- “apps” folder with programs and documentation to be installed in /Applications/TeX
- File “buildPackage.sh”
- Files “background.jpg”, “License.rtf”, “ReadMe.rtf”, “Welcome.rtf”
- “TeX-Source” folder containing LaTeX source files for the documentation in apps
- script folder containing “postinstall”
- GUIApplicationsBuild project for Packagemaker
- “TUG” folder with this document
- “OldStyle” folder with files used to build an old style legacy GUI-Applications.pkg

## 2. OVERVIEW

This package installs BibDesk, LaTeXiT, TeX Live Utility, TeXShop, and TeXworks in /Applications/Utilities. It installs the spell checker Excalibur. It also installs brief documentation for new features in that year’s TeX Live. All these pieces are added to the apps folder before building. They are moved to root/Applications/TeX by buildPackage.sh. Notice that buildPackage.sh must be edited if file names change or documentation is added or removed.

---

*Date:* April 13, 2013.

### 3. PROGRAMS IN APPS

To save space, the “apps” directory in the Build documentation tree only contains documentation, but does not contain GUI applications. Thus before building, it is necessary to add Applications to this directory. The following is a list of these applications, and of web sites where the current versions can be obtained:

- BibDesk, available at <http://bibdesk.sourceforge.net/>
- Excalibur-4.0.7, available at <http://excalibur.sourceforge.net/>. This is a folder named “Excalibur-4.0.7” containing the Excalibur program, Excalibur-manual.pdf, gpl-2.0.pdf, Read Me.txt, and Standard Dictionary.
- LaTeXiT, available at [http://chachatelier.fr/programmation/latexit\\_en.php](http://chachatelier.fr/programmation/latexit_en.php)
- TeX Live Utility, available at <http://mactlmgr.googlecode.com/>
- TeXShop, available at <http://pages.uoregon.edu/texshop/> The folder contains two programs named TeXShop and TeXShop-Lion. The first is a universal binary, and the second has 64 bit code and requires Lion or higher.
- TeXWorks, available at <http://tug.org/texworks/>. The folder contains two programs named TeXworks and TeXworks-SnowLeopard. These were just recently released. The first is a universal binary. The second has 64 bit code for Snow Leopard and higher.
- FixMacTeX2013.pkg. This is a small install package which fixes problems when a system containing MacTeX is upgraded to a later version of the operating system.

### 4. BUILDING

The first step in building a new package is to update all the supplied programs. Bibdesk, TeX Live Utility, and TeXShop all use the Sparkle mechanism to update, so it suffices to read these programs and select the “Check for Updates...” menu. LaTeXit may use the Sparkle mechanism as well, but in that case it seems to gray out the “Check for Updates...” menu when the program is up to date. To be sure, check

[http://chachatelier.fr/programmation/latexit\\_en.php](http://chachatelier.fr/programmation/latexit_en.php)

TeXworks updates can be found on [www.tug.org/texworks](http://www.tug.org/texworks).

The Excalibur spell checker is available at <http://excalibur.sourceforge.net/>. Note that it is provided in a folder containing the version number. If a new version is available, be sure to revise buildPackage.sh so it will copy the new folder to root/Applications/TeX.

Then update the documentation. Note that source code for the documentation is in the TeX-Source folder.

Finally, build by

```
sudo sh buildPackage.sh
```

## 5. DOCUMENTATION

Since Apple’s installer cannot install in user directories, there is no convenient place for it to install documentation. But sometimes new features of a release need to be pointed out. In that case, we install such documents in `/Applications/TeX`.

In general, it is a bad idea to put extensive documentation in `/Applications/TeX` because TeX Live has extensive and up to date information. Thus documentation we add tends to duplicate TeX Live documentation, and also tends not to be up to date. Currently, we provide the following documents:

- READ ME FIRST: This document is designed to help new users begin using the system
- LuaTeX-and-XeTeX: This document was added in 2010 to show users how to use system fonts in LuaTeX, and remind them that XeTeX has the same ability.
- Eps-and-PdfLaTeX: This was added to explain that TeX Live 2010 automatically converted eps files to pdf files when running pdfTeX.
- What is installed: An explanation of what is installed and where. Particularly useful when people want to remove things.

## 6. UPDATING FOR A NEW YEAR

Since the package name does not change from year to year, very little editing is require. The “Welcome.rtf” should be edited to reflect the release date.

## 7. BUILDING AN OLD STYLE GUI-APPLICATIONS.PKG FILE

Change to the directory “OldStyle” and issue the commands

```
sudo sh buildOldStyle.sh
```

## 8. BUILDING THE FINAL INSTALL PACKAGE WITH PACKAGEMAKER

The remaining steps are optional and not required for the final build of MacTeX.

Now edit `GUIApplicationsBuild`, the Packagemaker project file, using the GUI interface, as follows.

- (1) Click the top item on the left. The right side will change to a view with three tabs. First select the Configuration tab. Edit so
  - (a) Title: GUI-Applications
  - (b) User Sees: Easy Install Only
  - (c) Install Destination: System volume
  - (d) Certificate: Leave this blank. You might think that putting a certificate name here would correctly sign the package, but experiments show that it does not. The package needs to be signed after it is constructed.
  - (e) Description: Empty
- (2) The Requirements and Actions tab entries for the top item can be left blank
- (3) Click the entry GUI-Applications in the Contents section on the left. The right side will change to a view with two tabs. Select the Configuration tab. Edit so
  - (a) Choice Name: GUI-Applications
  - (b) Identifier: choice1
  - (c) Initial State: Selected and Enabled
  - (d) Destination: Leave Blank
  - (e) Tooltip: Leave Blank
  - (f) Description: Leave Blank
- (4) The Requirements tab entries for this contents item can be left blank
- (5) Open the GUI-Applications entry in the bottom left column. This item should show an “Applications” directory, obtained by dragging root/Applications to the left column. The right side will change to a view with four tabs. It is not necessary to edit the items under the Contents tab. But it is *very important* to edit items in the Components tab, turning off all boxes labeled “Allow Relocation”. The items under the Configuration tab should be edited as follows:
  - (a) Install: root/Applications (The item on the right should have selected “Relative To Project”)
  - (b) Destination: /Applications (The item on the right should have selected “Relative To Project”)
  - (c) Do not check “Allow custom location”
  - (d) Patch: No patch root selected
  - (e) Package Identifier: org.tug.mactex.gui

- (f) Package Version: 1.0
- (g) Restart Action: None
- (h) Check “Require admin authentication”
- (i) PackageLocation: Self-Contained

The items under the Scripts tab should be edited to read

- (a) Scripts directory: scripts (The item on the right should have selected “Relative To Project”)
- (b) Preinstall: Leave Blank
- (c) Postinstall: scripts/postinstall (The item on the right should have selected “Relative To Project”)

After verifying all these items, build the install package by clicking the “Build” icon. When asked to select a name, choose “GUI-Applications-Temp”. After the package has been made, sign it by executing the following line in Terminal:

```
sh signPackage.sh
```

This script simply calls

```
productsign --sign "Developer ID Installer: Richard Koch" old-name new-name
```

where “old-name” is GUI-Applications-Temp.pkg and “new-name” is GUI-Applications.pkg.

Packagemaker will report a large number of warnings. A few of these are incorrect permissions for links, but I believe these permissions are correct. The majority of the remaining warnings say “Group ID 80 does not match 20 of root volume.” Experimentation involving changing the groups of these applications changes this to “Group ID 20 does not match 80 of root volume.” I do not believe this warning is important, but it would be nice to know more about it.

Further investigation suggests that the error might be caused by hidden .DS\_Store folders. Search for these in root before building.