

TUG-TEXLIVE-2011

RICHARD KOCH

1. TEXLIVE-2011

This is the portion of the TUG documentation which deals with making the TeXLive-2011 and TeXLive-2011-DVD packages. This is the key component of the build directory.

A clean package file will contain

- Files “buildPackage.sh”, “buildPackage1.sh”, “buildPackageDVD.sh”, “buildPackageDVD1.sh”
- Files “Description.plist”, “Info.plist”, and “InfoGood.plist”
- “DVD” folder containing files “Description.plist”, “Info.plist”, “texlive.profile”, “texlivea4.profile”, and a folder named “resources” with various files and an English.lproj folder
- “PreferencePane” folder containing the “TeX Distribution” Preference Pane
- “resources” folder, containing various files and a folder named English.lproj with other files
- “Tools” folder, containing the original png background image later converted to jpg and used in the package.
- “TUG” folder with this document
- “xzDVD” folder, used to make an xz file with the package to submit to TUG; this file contains the xz Universal Binary and an xzscript

2. UPDATING FOR A NEW YEAR

To update the two packages for a new year, edit “buildPackage.sh”, “buildPackage1.sh”, “buildPackageDVD.sh”, and “buildPackageDVD1.sh” to reflect the new package names, which contain the year. Also update Description.plist, Info.plist, DVD/Description.plist, and DVD/Info.plist to reflect the new year.

Also edit the License.rtf, ReadMe.rtf, and Welcome.rtf files in resources/English.lproj/ to reflect the new name and new release date, and similarly edit DVD/resources/English.lproj.

Then edit License.rtf, ReadMe.rtf, and Welcome.rtf in DVD/resources/English.lproj for the same reason.

Then edit buildPackage.sh, buildPackageDVD.sh, and also

```
resources/postflight and DVD/resources/postflight
```

to reflect the appropriate year; scan these files to check that nothing else has changed. For instance, to change from 2010 to 2011, use Search and Replace to change the many, many occurrences of 2010 to 2011 in these files.

3. CREATING THE TWO PACKAGES

Now comes the central step. Issue the commands

```
cd /usr
sudo mv local local-temp
```

Then make sure local configuration files for TeX Live are disabled, perhaps by writing

```
cd
cd Library
mv texmf texmfold
rm -R texlive
```

Then install TeX Live 2011 using the TeX Live install script. First make sure that the local copy of the svn is up to date. I download the svn to a folder named texlive2011, and I'll assume that is the name of the svn folder. To update

```
cd
cd texlive2011
rsync -a --exclude=.svn --delete --delete-excluded tug.org::tldevsrc .
```

If building has just begun, new versions of the TeX Live binaries for OS X may have just been built. In that case it is necessary to replace old binaries in the svn with the new copies (which presumably have also been submitted to TUG). To do that, trash the old files in texlive2011/Master/bin/universal-darwin and texlive2011/Master/bin/x86_64-darwin, because if a binary used to be part of TeX Live 2011 but isn't any longer, we don't want it to show up in the install package. Then copy the contents of Binary/universal-darwin to texlive2011/Master/bin/universal-darwin. Copy the contents of Binary/x86_64-darwin to texlive2011/Master/bin/x86_64-darwin.

After OS X binaries are stable, these steps are no longer needed.

Then install TeX Live from the svn directory by

```

TEXLIVE_INSTALL_NO_CONTEXT_CACHE=1
export TEXLIVE_INSTALL_NO_CONTEXT_CACHE
cd
cd texlive2011/Master
sudo env TEXLIVE_INSTALL_NO_CONTEXT_CACHE=$TEXLIVE_INSTALL_NO_CONTEXT_CACHE ./install-t

```

The install script will present a short menu of configuration options. Modify defaults as follows:

- Use the “B” option to select both universal-darwin and x86_64-darwin binaries.
- Use the “D” option to change some directories:
 - Use the “5”, TEXMFVAR, option to select


```
~/Library/texlive/2011/texmf-var
```
 - Use the “6”, TEXMFCONFIG, option to select


```
~/Library/texlive/2011/texmf-config
```
 - Use the “7”, TEXMFHOME, option to select


```
~/Library/texmf
```
- Use the “O” option to select “use letter size instead of A4 by default”

Then start installation.

When installation is complete, switch the default tlmgr repository to the final value using the command (assuming that the TeXLive-2011 install package is made on an Intel system)

```

sudo /usr/local/texlive/2011/bin/universal-darwin/tlmgr option repository \
http://mirror.ctan.org/systems/texlive/tlnet

```

When installation is complete, go to

```
/usr/local/texlive/2011/tlpkg
```

and copy the file texlive.profile to the DVD build directory inside this TeXLive-2011 build folder, replacing the current copy of texlive.profile there. This profile will be added to the DVD install package, to be used when TeX Live is installed from the independent copy on the TUG DVD.

In addition, duplicate `texlive.profile` in the DVD directory, calling the new copy `texlivea4.profile`. Then edit this new file, changing “`option_letter 1`” to “`option_letter 0`” near the end of the file.

After these steps, the final TeXLive install packages can be built, using the commands:

```
sudo sh buildPackage.sh
sudo sh buildPackageDVD.sh
```

Finally

```
cd
cd /usr
sudo rm -R local
sudo mv local-temp local
```

4. MAKING THE XZ FILE

Put `TeXLive-2011-DVD.mpkg` in the `xzDVD` folder. Then create the `xz` file using the command

```
sh xzscript
```

5. ADJUSTING THE TEXLIVE-2011 AND TEXLIVE-2011-DVD INSTALL PACKAGES

The folder `resources/English.lproj` contains some files which are shown to the user during installation. One of these files is a brief Welcome document, a second is a longer ReadMe file, and a third shows the License for the code. After the TeXLive install packages are constructed, you may need to revise these documents. For example, the Welcome document contains the release date. Usually TeX Live authors are overly ambitious and predict a release several months earlier than it actually occurs, and you’ll need to change the release date from time to time.

Two special scripts are provided to make changes in these support files without rebuilding everything. The scripts are called `buildPackage1.sh` and `buildPackageDVD1.sh`. The `buildPackage1.sh` script uses the already constructed root folder, and makes a new install package using this root, and possible new versions of `Description.plist`, `Info.plist`, and the documents in `resources/English.lproj`. The `buildPackageDVD1.sh` script works similarly, using new versions inside the DVD folder of `Description.plist`, `Info.plist` and documents in `DVD/resources/English.lproj`.