

TUG-BINARY

RICHARD KOCH

1. TEX LIVE BINARIES

This is the portion of the TUG documentation which deals with making the TeX Live binaries. Building is a two stage process. First TeX Live and asymptote are compiled on Intel Leopard, PowerPC Leopard, and Intel Snow Leopard. Then the resulting binaries are processed in this directory, forming the universal binaries sent to TUG for inclusion in TeX Live.

A clean package file will contain

- The folder BuildingBinaries, containing a file named “Build Status on Intel and PPC”, and a folder named “AsymptoteBuild” with files Test and Test.png.
- An empty folder named “RawCodeFromCompile”
- “liposcript-universal” and “liposcript-x86_64”
- “TUG” folder with this document
- xindy-2010 folder with several xindy.mem and xindy.run files

In addition, the main MacTeX folder containing Binary must contain “xz”, the binary which compresses packages.

2. COMPILING THE BINARIES

The “BuildingBinaries” folder contains a crucial document named

Build Status on Intel and PPC.rtf

This document contains the precise shell commands needed to build TeX Live binaries on Intel Leopard, PowerPC Leopard, and Intel Snow Leopard. The document is an rtf file rather than a TeX file so it can be easily edited. Any change in compile commands should immediately be added to the document. I find it convenient to put a copy of the document on the desktop of each machine, merging editing changes back into the master document from time to time.

Building xindy requires obtaining and compiling CLISP. The document explains exactly how that is done. It only needs to be done once on each platform.

Building asymptote requires obtaining, compiling, and installing three libraries. This is also explained in detail in the document. The source code for these libraries should be placed in a folder named AsymptoteBuild in the home directory of each machine. This folder should contain two additional files: Test, and Test.png. A sample folder with these files is in the BuildingBinaries folder.

The end result of this process on Intel Leopard will be two files on the desktop:

- i386-darwin.tar.gz
- asy-Intel

The end result on Intel Snow Leopard will be two files on the desktop:

- x86_64-darwin.tar.gz
- asy-x86_64

The end result on PowerPC Leopard will be two files on the desktop:

- powerpc-darwin.tar.gz
- asy-PPC

Move all of these files to the Binary/RawCodeFromCompile directory of the MacTeX build system on the machine used to make final packages. Unzip the three zip files to form three subfolders of Binary/RawCodeFromCompile named i386-darwin, powerpc-darwin, and x86_64-darwin.

3. MAKING UNIVERSAL BINARIES

To build the universal-darwin binary directory, go to the Binary directory and execute the command

```
sudo sh liposcript-universal
```

This will create a folder named universal-darwin and a file named universal-darwin.tar.xz. Send the xz file to Karl Berry.

To build the x86_64-darwin binary directory, go to the Binary directory and execute the command

```
sudo sh liposcript-x86_64
```

This will create a folder named x86_64-darwin and a file named x86_64-darwin.tar.xz. Send the xz file to Karl Berry.

4. XINDY

In 2011, due to a strange interaction between Xindy and CLisp, the `xindy.run` and `xindy.mem` files are corrupt. But the 2010 versions of these files work, and the files have not been changed. So the two liposcripts replace 2011 versions of the files with the 2010 versions. In universal-darwin life is a little more complicated because `xindy.mem` is not processor independent, but the liposcripts still do the right thing.