About BasicTeX-2022

Richard Koch

January 11, 2022

1 Introduction

Most TeX distributions for Mac OS X are based on TeX Live, the reference edition of TeX produced by TeX User Groups across the world. Among these is MacTeX, which installs the full TeX Live as well as front ends, Ghostscript, and other utilities — everything needed to use TeX on the Mac. To obtain it, go to `http://tug.org/mactex`.

2 Basic TeX

BasicTeX (92 MB) is an installation package for Mac OS X based on TeX Live 2022. Unlike MacTeX, this package is deliberately small. Yet it contains all of the standard tools needed to write TeX documents, including TeX, LaTeX, pdfTeX, MetaFont, dvips, MetaPost, and XeTeX.

It would be dangerous to construct a new distribution by going directly to CTAN or the Web and collecting useful style files, fonts and so forth. Such a distribution would run into support issues as the creators move on to other projects. Luckily, the TeX Live install script has its own notion of "installation packages" and collections of such packages to make "installation schemes." BasicTeX is constructed by running the TeX Live install script and choosing the "small" scheme. Thus it is a subset of the full TeX Live with exactly the TeX Live directory structure and configuration scripts. Moreover, BasicTeX contains tlmgr, the TeX Live Manager software introduced in TeX Live 2008, which can install additional packages over the network. So it will be easy for users to add missing packages if needed.

Since it is important that the install package come directly from the standard TeX Live distribution, I'm going to explain exactly how I installed TeX to produce the install package. First I renamed `/usr/local/texlive` so the TeX Live installer would create a virgin copy of "texlive" in this location. Then I installed the small scheme from TeX Live using the TeX Live install script. Then I copied /usr/local/texlive to a folder named "root" in the

directory where the install package was made. Apple's PackageMaker then created the install package from this "root" folder.

The latest version of TeX Live 2022 can be obtained over the web by going to `http://tug.org/texlive/svn/`. The "rsync" method listed first will download everything needed without using subversion.

To install TeX Live, issue the following commands:

```
cd "directory containing texlive"
sudo ./install-tl
```

This will run the TeX Live install script, a straightforward script (kudos to the author). Do not accept the default choices; instead make the following changes before installing:

- Use the "S" option to choose the "TeX Live small scheme"

- Use the "D" option to change the TEXDIR from /usr/local/texlive/2022 to

    `/usr/local/texlive/2022basic`

- Use the "D" option to change TEXMFHOME from $HOME/texmf to

    `$HOME/Library/texmf`

    This is the local directory where users can add their own style files, fonts, bibtex files, etc., without running texhash. Most of the current TeX distributions for Mac OS X use this location.

- Similarly, use "D" to change TEXMFVAR to

    `$HOME/Library/texlive/2022basic/texmf-var`

    and TEXMFCONFIG to

    `$HOME/Library/texlive/2022basic/texmf-config`

- use the "O" option to tell the script not to install the macro/font doc tree or the macro/font source tree, to avoid building fmt files, and to configure for letter size paper (this is later changed by the installer's postflight script to the user's default paper size).

Then install.

Reset the default location where tlmgr looks for updates as follows:

```
sudo tlmgr option location http://mirror.ctan.org/systems/texlive/tlnet
```

To get a working TeX, we need only configure papersize, make formats, and perform other initialization commands. The following commands would do the trick; here $PAPER is either "letter" or "a4".

```
sudo bash
PATH=/usr/local/texlive/2022basic/bin/universal-darwin:$PATH; export PATH
texhash
sudo tlmgr paper $PAPER
sudo fmtutil-sys --all
```

Actually, we make the install package without running these commands, and then run them in the postinstall script. Thus the install package itself does not contain format files, which are built at the end of the install process. The postinstall script determines the user's default paper size and configures TeX appropriately.

3   Testing These Packages

The 2022 version of the package runs on Mojave, Catalina, and Big Sur, and should run on Apple's next version of macOS to be introduced in the fall of 2022. This is in line with a new support policy for MacTeX. We now support all versions of macOS still receiving security updates from Apple, and thus the last three versions of the system. BasicTeX-2022 has universal binaries with both Intel and Arm code.

4   Worrying About Europe

Donald Knuth wrote the first version of TeX in 1978, and rewrote the program using Tangle and Weave in 1982. He wrote a final third version in 1989, mainly to increase font tables from 128 characters to 256 characters for better foreign language support. The current TeX is version 3.14159265.

From the beginning, TeX could typeset words with diacritical marks using escape sequences, but these escape sequences separated the word internally into pieces and hyphenation tables broke for Europeans. The move to 256 character fonts was designed to fix this problem. Accented characters are now characters in their own right and hyphenation tables can account for them.

Using this facility requires new fonts to replace Knuth's computer modern fonts. The BasicTeX package contains the original computer modern fonts (in Adobe type one format) and new Latin Modern fonts by Boguslaw Jackowski and Janusz M. Nowacki for European users. The creation of these very compact outline fonts is a relatively new development. BasicTeX also contains the babel and inputenc packages for dealing with these European characters. One of our fears has been that we would not be able to support European

users with such a small package, but the Latin Modern package makes this support possible.

Readers who want to know more should consult Jackowski and Nowacki's article from EuroTeX2005: `www.dante.de/dante/events/eurotex/papers/TUT09.pdf`.

5   The PostInstall Script

TeX users on the Mac are often not familiar with Unix. The BasicTeX install package does not leave these users to fend for themselves; instead it configures TeX so it is ready to use. After installing TeX, a user need only download a suitable front end and begin working; many of these front ends will automatically configure themselves for BasicTeX.

The install package contains a postinstall script. This script does three things:

- It uses Apple routines to find the user's default papersize, and configures TeX for that papersize.

- It modifies the PATH and MANPATH variables so they correctly point to the new TeX

- It installs a "TeX Distribution" data structure by Gerben Wierda and Jerome Laurens in `/Library/TeX`.

I'll explain these operations in the following sections.

It is useful to know exactly where MacTeX installs files so there are no surprises later on. Here is a list:

/usr/local/texlive/2022basic	the actual TeX distribution
/Library/TeX/texbin	a symbolic link to the TeX binaries
/Library/TeX	the TeX distribution structure, mainly symbolic links

6   Setting Paper Size

When the TeX Live installation script finishes, it prints a message suggesting that the user configure papersize and reset PATH and MANPATH shell configuration variables by hand using command line arguments. This isn't in the spirit of the Mac.

Since it is easy for a script to read off the user's default papersize, the first of these steps can easily be done automatically.

```
sudo tlmgr paper letter
```

where "letter" will be "a4" in many countries.

7 Configuring PATH and MANPATH

On modern Macs, the global shell initialization scripts `/etc/csh.login` and `/etc/profile` call the helper script `/usr/libexec/path_helper`, which initializes PATH with the contents of `/etc/paths` and initializes MANPATH with the contents of `/etc/manpath`. Then it extends PATH with any additional paths listed in files within `/etc/paths.d` and extends MANPATH with additional manpaths listed in files within `/etc/manpaths.d`.

The path `/usr/local/bin` is one of the paths in `/etc/paths`, so it is not necessary to add this path when installing ghostscript and other programs. In the shipping version of Leopard, `/etc/paths.d` and `/etc/manpaths.d` each contain a single file named X11, and so PATH and MANPATH include entries for X11.

When the BasicTeX package installs, it adds a file named TeX to `/etc/paths.d` containing the single entry

    /Library/TeX/texbin

and adds a file named TeX to `/etc/manpaths.d` containing the single entry

     /Library/TeX/Distributions/.DefaultTeX/Contents/Man

Thus TeX programs can be run from the command line, and TeX man pages can be read from the command line.

8 The TeX Distribution Data Structure

As mentioned earlier, BasicTeX-2022 installs a data structure by Gerben Wierda and Jerome Laurens to support multiple TeX distributions on a machine. This data structure is also by the full versions of MacTeX.

The data structure knows about almost all TeX distributions used on Mac OS X today, including the TeXLive distributions from 2006 through 2022. The data structure can easily be extended to cover future distributions.

None of these distributions are written to the same location, so a user may have multiple distributions installed. The GUI program *TeX Live Utility*, available for free on the internet, can update and maintain BasicTeX-2022. It has a menu item called "Change Default TeX Live Version" which lists all active TeX distributions on the computer and allows the user to select the active one by clicking a button. Changing distributions this way automatically changes PATH and MANPATH variables, so interaction with TeX via the command line will use the appropriate distribution. It also automatically reconfigures GUI applications so they use the new active distribution. To make this happen, a symbolic link named `/Library/TeX/texbin` has been created, pointing idirectly to the binary directory

of the active distribution. Some GUI applications will need to be reconfigured to use this link, but after they are configured, they never need be touched again.

Why is this structure useful? Here is an example. Last year's TeX Live distribution installed in `/usr/local/texlive/2020`. This year's distribution installs in `usr/local/texlive/2022`. Users can confidently install the 2022 distribution without worrying that it will break their current projects. If there are problems with the 2022 version, they can return to the 2020 version with a single click. When experiments show that the new distribution causes no problem, users can make it permanently active.

Another example: users who already have TeX can install BasicTeX to test it without worrying that it will overwrite their current distribution.

A more important example: suppose a new user installs BasicTeX, uses it more and more, and then wants a more complete TeX. At the TUG MacTeX site, they can find a similar install package for the complete TeX Live, which is 30 times larger than BasicTeX. This package installs exactly as BasicTeX. Neither of these distributions overwrite each other, and the user can easily switch back and forth until they are confident that the larger distribution works.

Incidentally, the data structure does not modify the actual TeX distribution in any way. It is ingeniously constructed so that TeX Live Utility needs to change only a single symbolic link to change the active TeX distribution. The data structure is very small, essentially a collection of symbolic links in `/Library/TeX`.

Users are free to use the data structure or ignore it. More and more front ends are likely to use it.

TeX distributions are installed in locations which are usually not displayed in the Finder. But it is possible to inspect the currently active distribution by going to `/Library/TeX/Root` with Finder. This is a symbolic link to the full currently active distribution. Similarly, `/Library/TeX/Documentation` is a symbolic link to all documentation folders for the currently active distribution.

The data structure contains features which future GUI programs may use. For example, a TeX interface program could offer to typeset some files with one distribution and others with another, or to display a list of documentation for the currently active distribution.

There is one complication. The TeX distribution data structure adds `/Library/TeX/texbin` to the end of the PATH variable. But Fink adds `/sw/bin` to the start of PATH and MacPorts adds `/opt/local/bin` to the start, so commands issued from the Terminal will find the TeX in Fink or MacPorts TeX regardless of the active TeX chosen by TeX Live Utility.. This only applies to users who have Fink or MacPorts. For these users, the problem is easily fixed. In the shell configuration script, which is probably .profile

or .bash_profile or .bash_login, modify PATH so `/Library/TeX/texbin` comes first. For a bash shell, the required commands at the end of the script should be

```
export PATH=/Library/TeX/texbin:$PATH
export MANPATH=/Library/TeX/Distributions/.DefaultTeX/Contents/Man:$MANPATH
```

9 XeTeX

We managed to get a lot of functionality into BasicTeX. It has all of the standard programs and style files that one would expect: TeX, pdfTeX, LaTeX, pdfLaTeX, AMS-LaTeX, metafont, dvips, xdvi, metapost, PStricks, beamer, RevTeX, etc. But our biggest coup is the inclusion of XeTeX, a version of TeX by Jonathan Kew.

XeTeX solves two of TeX's biggest problems: its inability to accept unicode input, and its dependence on special TeX fonts and inability to use system fonts. XeTeX first modifies TeX so it accepts source files written in UTF-8 format. Then it modifies TeX's font routines so that instead of consulting TeX Font Metric files to learn about character dimensions, kerning pairs, and the like, the program directly calls system font routines to find this information. The result is that XeTeX can use system fonts without preliminary processing. The program can mix these system fonts with ordinary TeX fonts; for instance mathematical formulas can be written with standard TeX fonts in a document using a system font for text.

The next four pages contain the source and output for two XeTeX examples. The first uses the Zapfino font. Jonathan would probably not approve of this example because he doesn't like gee-whiz demonstrations, preferring to demonstrate XeTeX with real work and fonts which do not distract from the presentation. Notice how few lines are needed to activate the mechanisms of XeTeX.

The real advantage of XeTeX becomes apparent in the second example, showing text in a non-Latin foreign language. Many front ends on the Mac are written with Cocoa and fully support unicode, International input, right to left text input, and so forth. This makes it possible to use TeX for documents written partly or entirely in Chinese, Hebrew, Arabic, and a host of other languages, and directly type source text in these languages.

```
% These lines tell TeXShop to typeset with xelatex, and to open
% and save the source with Unicode encoding.

%!TEX TS-program = xelatex
%!TEX encoding = UTF-8 Unicode

\documentclass[12pt]{article}
\usepackage{geometry}
\geometry{letterpaper}
\usepackage{graphicx}
\usepackage{amssymb}

% Will Robertson's fontspec.sty can be used to simplify font choices.
% To experiment, open /Applications/Font Book to examine the fonts
% provided on Mac OS X, and change "Zapfino" to any of these choices.

\usepackage{fontspec,xltxtra,xunicode}
\defaultfontfeatures{Mapping=tex-text}
\setromanfont[Mapping=tex-text]{Zapfino}
\setsansfont[Scale=MatchLowercase,Mapping=tex-text]{Gill Sans}
\setmonofont[Scale=MatchLowercase]{Andale Mono}

\title{Brief Article}
\author{The Author}

\begin{document}
\maketitle

This is a standard \TeX\ document using Zapfino type. It uses \LaTeX\ and
the standard \TeX\ graphic packages.

\begin{figure}[htbp]
  \centering
  \includegraphics[width=2in]{example.pdf}
  \caption{from Mathematica}
\end{figure}

\end{document}
```
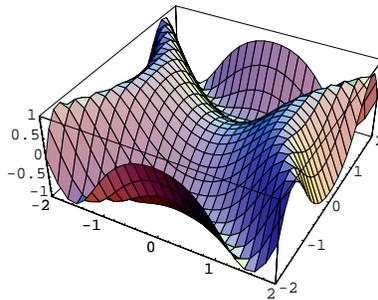
Brief Article

The Author

January 11, 2022

This is a standard TEX document using Zapfino type.

It uses LATEX and the standard TEX graphic packages.

Figure 1: from Mathematica

% These lines tell TeXShop to typeset with xelatex, and to open and
% save the source with Unicode encoding.

%!TEX TS-program = xelatex
%!TEX encoding = UTF-8 Unicode

\documentclass[12pt]{article}
\usepackage{geometry}
\geometry{letterpaper}
\usepackage{graphicx}
\usepackage{amssymb}

% Will Robertson's fontspec.sty can be used to simplify font choices.
% To experiment, open /Applications/Font Book to examine the fonts
% provided on Mac OS X, and change "Hoefler Text" to any of these choices.

\usepackage{fontspec,xltxtra,xunicode}
\defaultfontfeatures{Mapping=tex-text}
\setromanfont[Mapping=tex-text]{Hoefler Text}
\setsansfont[Scale=MatchLowercase,Mapping=tex-text]{Gill Sans}
\setmonofont[Scale=MatchLowercase]{Andale Mono}

\title{Brief Article}
\author{The Author}

\begin{document}
\maketitle

This is a standard TeX document using Hoefler Text. It uses \LaTeX\ and
the standard \TeX\ graphic packages.

% For many users, the previous commands will be enough.
% If you want to directly input Unicode, add an Input Menu or Keyboard to the menu bar
% using the International Panel in System Preferences.
% Unicode must be typeset using a font containing the appropriate characters.

\newfontfamily{\A}{Geeza Pro}
\newfontfamily{\HAA}[Scale=0.9]{Lucida Grande}
\newfontfamily{\J}[Scale=0.85]{Osaka}

Here are some multilingual Unicode fonts: this is Arabic text: {\A السلام عليكم},
this is Hebrew: {\HAA שלום}, and here's some Japanese: {\J 今日は}.

\end{document}

Brief Article

The Author

January 11, 2022

This is a standard TeX document using Hoefler Text. It uses LaTeX and the standard TeX graphic packages.

Here are some multilingual Unicode fonts: this is Arabic text: السلام عليكم, this is Hebrew: שלום, and here's some Japanese: 今日は.