The TEX Live Guide—2015

Karl Berry, editor

http://tug.org/texlive/

May 2015

Contents

1 Introduction **2**
1.1 TEX Live and the TEX Collection . . . 2
1.2 Operating system support . . . 3
1.3 Basic installation of TEX Live . . . 3
1.4 Security considerations . . . 3
1.5 Getting help . . . 3

2 Overview of TEX Live **4**
2.1 The TEX Collection: TEX Live, proTEXt, MacTEX . . . 4
2.2 Top level TEX Live directories . . . 4
2.3 Overview of the predefined texmf trees . . . 5
2.4 Extensions to TEX . . . 6
2.5 Other notable programs in TEX Live . . . 6

3 Installation **7**
3.1 Starting the installer . . . 7
3.1.1 Unix . . . 7
3.1.2 Mac OS X . . . 8
3.1.3 Windows . . . 8
3.1.4 Cygwin . . . 8
3.1.5 The text installer . . . 9
3.1.6 The expert graphical installer . . . 9
3.1.7 The simple wizard installer . . . 9
3.2 Running the installer . . . 9
3.2.1 Binary systems menu (Unix only) . . . 10
3.2.2 Selecting what is to be installed . . . 10
3.2.3 Directories . . . 11
3.2.4 Options . . . 12
3.3 Command-line install-tl options . . . 13
3.3.1 The `-repository` option . . . 14
3.4 Post-install actions . . . 14
3.4.1 Environment variables for Unix . . . 14
3.4.2 Environment variables: Global configuration . . . 15
3.4.3 Internet updates after DVD installation . . . 15
3.4.4 System font configuration for XeTEX and LuaTEX . . . 15
3.4.5 ConTEXt Mark IV . . . 15
3.4.6 Integrating local and personal macros . . . 16
3.4.7 Integrating third-party fonts . . . 16
3.5 Testing the installation . . . 16
3.6 Links for additional downloadable software . . . 18

4 Specialized installations **18**
4.1 Shared-user (or cross-machine) installations . . . 18
4.2 Portable (USB) installations . . . 19
4.3 ISO (or DVD) installations . . . 19

5 tlmgr: Managing your installation 19
5.1 tlmgr GUI mode . . . 20
5.2 Sample tlmgr command-line invocations . . . 20

6 Notes on Windows 22
6.1 Windows-specific features . . . 22
6.2 Additional software included on Windows . . . 22
6.3 User Profile is Home . . . 23
6.4 The Windows registry . . . 23
6.5 Windows permissions . . . 23
6.5.1 Protecting the TEX Live directory . . . 24
6.6 Increasing maximum memory on Windows and Cygwin . . . 24

7 A user's guide to Web2C 24
7.1 Kpathsea path searching . . . 25
7.1.1 Path sources . . . 25
7.1.2 Config files . . . 26
7.1.3 Path expansion . . . 26
7.1.4 Default expansion . . . 27
7.1.5 Brace expansion . . . 27
7.1.6 Subdirectory expansion . . . 27
7.1.7 List of special characters and their meaning: a summary . . . 27
7.2 Filename databases . . . 28
7.2.1 The filename database . . . 28
7.2.2 kpsewhich: Standalone path searching . . . 28
7.2.3 Examples of use . . . 29
7.2.4 Debugging actions . . . 30
7.3 Runtime options . . . 32

8 Acknowledgements 33

9 Release history 34
9.1 Past . . . 34
9.1.1 2003 . . . 35
9.1.2 2004 . . . 35
9.1.3 2005 . . . 37
9.1.4 2006–2007 . . . 37
9.1.5 2008 . . . 38
9.1.6 2009 . . . 38
9.1.7 2010 . . . 39
9.1.8 2011 . . . 39
9.1.9 2012 . . . 40
9.1.10 2013 . . . 40
9.1.11 2014 . . . 40
9.2 Present—2015 . . . 41
9.3 Future . . . 42

1 Introduction

1.1 TEX Live and the TEX Collection

This document describes the main features of the TEX Live software distribution — TEX and related programs for GNU/Linux and other Unix flavors, Mac OS X, and Windows systems.

You may have acquired TEX Live by downloading, or on the TEX Collection DVD, which TEX user groups distribute among their members, or in other ways. Section 2.1 briefly describes the contents of the DVD. Both TEX Live and the TEX Collection are cooperative efforts by the TEX user groups. This document mainly describes TEX Live itself.

TEX Live includes executables for TEX, LATEX 2ε, ConTEXt, METAFONT, MetaPost, BIBTEX and many other programs; an extensive collection of macros, fonts and documentation; and support for typesetting in many different scripts from around the world.

For a brief summary of the major changes in this edition of TeX Live, see the end of the document, section 9 (p. 34).

1.2 Operating system support

TeX Live contains binaries for many Unix-based platforms, including GNU/Linux, Mac OS X, and Cygwin. The included sources can be compiled on platforms for which we do not provide binaries.

As to Windows: Windows XP and later are supported. Windows 2000 will probably still mostly work. There are no special 64-bit executables for Windows, but the 32-bit executables should run on 64-bit systems.

See section 2.1 for alternate solutions for Windows and Mac OS X.

1.3 Basic installation of TeX Live

You can install TeX Live either from DVD or over the Internet (http://tug.org/texlive/acquire.html). The net installer itself is small, and downloads everything requested from the Internet.

The DVD installer lets you install to a local disk. You cannot run TeX Live directly from the TeX Collection DVD (or its `.iso` image), but you can prepare a runnable installation on, e.g., a USB stick (see section 4.2). Installation is described in later sections (p. 7), but here is a quick start:

- The installation script is named `install-tl`. It can operate in a "wizard mode" given the option `-gui=wizard` (default for Windows), a text mode given `-gui=text` (default for everything else), and an expert GUI mode given `-gui=perltk`.
- One of the installed items is the 'TeX Live Manager' program, named tlmgr. Like the installer, it can be used in both GUI mode and in text mode. You can use it to install and uninstall packages and do various configuration tasks.

1.4 Security considerations

To the best of our knowledge, the core TeX programs themselves are (and always have been) extremely robust. However, the contributed programs in TeX Live may not reach the same level, despite everyone's best efforts. As always, you should be careful when running programs on untrusted input; for maximum safety, use a new subdirectory.

This need for care is especially urgent on Windows, since in general Windows finds programs in the current directory before anything else, regardless of the search path. This opens up a wide variety of possible attacks. We have closed many holes, but undoubtedly some remain, especially with third-party programs. Thus, we recommend checking for suspicious files in the current directory, especially executables (binaries or scripts). Ordinarily they should not be present, and definitely should not normally be created by merely processing a document.

Finally, TeX (and its companion programs) are able to write files when processing documents, a feature that can also be abused in a wide variety of ways. Again, processing unknown documents in a new subdirectory is the safest bet.

1.5 Getting help

The TeX community is active and friendly, and most serious questions end up getting answered. However, the support is informal, done by volunteers and casual readers, so it's especially important that you do your homework before asking. (If you prefer guaranteed commercial support, you can forgo TeX Live completely and purchase a vendor's system; http://tug.org/interest.html#vendors has a list.)

Here is a list of resources, approximately in the order we recommend using them:

Getting Started If you are new to TeX, the web page http://tug.org/begin.html gives a brief introduction to the system.

TeX FAQ The TeX FAQ is a huge compendium of answers to all sorts of questions, from the most basic to the most arcane. It is included on TeX Live in `texmf-dist/doc/generic/FAQ-en/`, and is available on the Internet through http://www.tex.ac.uk/faq. Please check here first.

TeX Catalogue If you are looking for a specific package, font, program, etc., the TeX Catalogue is the place to look. It is a huge collection of all TeX-related items. See http://mirror.ctan.org/help/Catalogue/.

TEX Web Resources The web page http://tug.org/interest.html has many TEX-related links, in particular for numerous books, manuals, and articles on all aspects of the system.

support archives Principal support forums include the Usenet newsgroup news:comp.text.tex, the mailing list texhax@tug.org, and http://tex.stackexchange.com, a collaboratively edited question and answer site. Their archives have years of past questions and answers for your searching pleasure, via http://groups.google.com/group/comp.text.tex/topics and http://tug.org/mail-archives/texhax, respectively. And a general web search, for example on http://google.com, never hurts.

asking questions If you cannot find an answer, you can post to `comp.text.tex` through Google or your newsreader, or to texhax@tug.org through email. But before you post, *please* read this FAQ entry, to maximize your chances of getting a useful answer: http://www.tex.ac.uk/cgi-bin/texfaq2html?label=askquestion. Also worth mentioning are the LATEX Community site at http://www.latex-community.org/ and its forum http://www.latex-community.org/forum/, and TEX StackExchange at http://tex.stackexchange.com/.

TEX Live support If you want to report a bug or have suggestions or comments on the TEX Live distribution, installation, or documentation, the mailing list is tex-live@tug.org. However, if your question is about how to use a particular program included in TEX Live, please write to that program's maintainer or mailing list. Often running a program with the `--help` option will provide a bug reporting address.

The other side of the coin is helping others who have questions. Both `comp.text.tex` and `texhax` are open to anyone, so feel free to join, start reading, and help out where you can.

2 Overview of TEX Live

This section describes the contents of TEX Live and the TEX Collection of which it is a part.

2.1 The TEX Collection: TEX Live, proTEXt, MacTEX

The TEX Collection DVD comprises the following:

TEX Live A complete TEX system to be installed to disk. Home page: http://tug.org/texlive/.

MacTEX for Mac OS X, this adds a native Mac OS X installer and other Mac applications to TEX Live. Home page: http://tug.org/mactex/.

proTEXt An enhancement of the MiKTEX distribution for Windows, proTEXt adds a few extra tools to MiKTEX, and simplifies installation. It is entirely independent of TEX Live, and has its own installation instructions. Home page: http://tug.org/protext/.

CTAN A snapshot of the CTAN repository (http://www.ctan.org/).

CTAN and protext do not follow the same copying conditions as TEX Live, so be careful when redistributing or modifying.

2.2 Top level TEX Live directories

Here is a brief listing and description of the top level directories in a TEX Live installation.

`bin` The TEX system programs, arranged by platform.

`readme-*.dir` Quick overview and useful links for TEX Live, in various languages, in both HTML and plain text.

`source` The source to all included programs, including the main Web2C-based TEX distributions.

`texmf-dist` The principal tree; see `TEXMFDIST` below.

`tlpkg` Scripts, programs and data for managing the installation, and special support for Windows.

In addition to the directories above, the installation scripts and `README` files (in various languages) are at the top level of the distribution.

For documentation, the comprehensive links in the top-level file `doc.html` may be helpful. The documentation for nearly everything (packages, formats, fonts, program manuals, man pages, Info files) is in `texmf-dist/doc`. You can use the `texdoc` program to find documentation wherever it is located.

This TEX Live documentation itself is in `texmf-dist/doc/texlive`, available in several languages:

- Czech/Slovak: `texmf-dist/doc/texlive/texlive-cz`
- German: `texmf-dist/doc/texlive/texlive-de`
- English: `texmf-dist/doc/texlive/texlive-en`
- French: `texmf-dist/doc/texlive/texlive-fr`
- Italian: `texmf-dist/doc/texlive/texlive-it`
- Polish: `texmf-dist/doc/texlive/texlive-pl`
- Russian: `texmf-dist/doc/texlive/texlive-ru`
- Serbian: `texmf-dist/doc/texlive/texlive-sr`
- Simplified Chinese: `texmf-dist/doc/texlive/texlive-zh-cn`

2.3 Overview of the predefined texmf trees

This section lists the predefined variables specifying the texmf trees used by the system, and their intended purpose, and the default layout of TEX Live. The command `tlmgr conf` shows the values of these variables, so that you can easily find out how they map to particular directories in your installation.

All of the trees, including the personal ones, should follow the TEX Directory Structure (TDS, http://tug.org/tds), with all its myriad subdirectories, or files may not be found. Section 3.4.6 (p. 16) describes this in more detail.

`TEXMFDIST` The tree which holds nearly all of the files in the original distribution—configuration files, scripts, packages, fonts, etc. (The main exception are the per-platform executables, which are stored in a sibling directory `bin/`.)

`TEXMFLOCAL` The tree which administrators can use for system-wide installation of additional or updated macros, fonts, etc.

`TEXMFHOME` The tree which users can use for their own individual installations of additional or updated macros, fonts, etc. The expansion of this variable dynamically adjusts for each user to their own individual directory.

`TEXMFCONFIG` The (personal) tree used by the utilities `texconfig`, `updmap`, and `fmtutil` to store modified configuration data.

`TEXMFSYSCONFIG` The (site-wide) tree used by the utilities `texconfig-sys`, `updmap-sys`, and `fmtutil-sys` to store modified configuration data.

`TEXMFVAR` The (personal) tree used by `texconfig`, `updmap` and `fmtutil` to store (cached) runtime data such as format files and generated map files.

`TEXMFSYSVAR` The (site-wide) tree used by `texconfig-sys`, `updmap-sys` and `fmtutil-sys`, and also by `tlmgr`, to store (cached) runtime data such as format files and generated map files.

`TEXMFCACHE` The tree(s) used by ConTEXt MkIV and LuaLATEX to store (cached) runtime data; defaults to `TEXMFSYSVAR`, or (if that's not writable), `TEXMFVAR`.

The default layout is:

system-wide root can span multiple TEX Live releases:

`2014` A previous release.

`2015` The current release.

`bin`

`i386-linux` GNU/Linux binaries

`...`

`universal-darwin` Mac OS X binaries

`win32` Windows binaries

`texmf-dist` `TEXMFDIST` and `TEXMFMAIN`

`texmf-var` `TEXMFSYSVAR`, `TEXMFCACHE`

`texmf-config` `TEXMFSYSCONFIG`

`texmf-local` `TEXMFLOCAL`, intended to be retained from release to release.

user's home directory (`$HOME` or `%USERPROFILE%`)

`.texlive2014` Privately generated and configuration data for a previous release.

`.texlive2015` Privately generated and configuration data for the current release.

`texmf-var` `TEXMFVAR`, `TEXMFCACHE`
`texmf-config` `TEXMFCONFIG`

`texmf` `TEXMFHOME` Personal macros, etc.

2.4 Extensions to TEX

Knuth's original TEX itself is frozen, apart from rare bug fixes. It is present in TEX Live as the program tex, and will remain so for the foreseeable future. TEX Live also contains several extended versions of TEX(also known as TEX engines):

ε-TEX adds a set of new primitives (related to macro expansion, character scanning, classes of marks, additional debugging features, and more) and the TEX--XET extensions for bidirectional typesetting. In default mode, ε-TEX is 100% compatible with ordinary TEX. See `texmf-dist/doc/etex/base/etex_man.pdf`.

pdfTEX builds on the ε-TEX extensions, adding support for writing PDF output as well as DVI, and many non-output-related extensions. This is the program invoked for most formats, e.g., etex, latex, pdflatex. Its web site is http://www.pdftex.org/. See `texmf-dist/doc/pdftex/manual/pdftex-a.pdf` for the manual, and `texmf-dist/doc/pdftex/manual/samplepdf/samplepdf.tex` for example usage of some of its features.

LuaTEX is the designated successor of pdfTEX, and is mostly (but not entirely) backward-compatible. It is also intended to be a functional superset of Aleph (see below), though technical compatibility is not intended. The incorporated Lua interpreter (http://www.lua.org/) enables elegant solutions for many thorny TEX problems. When called as `texlua`, it functions as a standalone Lua interpreter, and is already used as such within TEX Live. Its web site is http://www.luatex.org/, and the reference manual is `texmf-dist/doc/luatex/base/luatexref-t.pdf`.

XeTEX adds support for Unicode input and OpenType- and system fonts, implemented using standard third-party libraries. See http://tug.org/xetex.

Ω **(Omega)** is based on Unicode (16-bit characters), thus supports working with almost all the world's scripts simultaneously. It also supports so-called 'Ω Translation Processes' (OTPs), for performing complex transformations on arbitrary input. Omega is no longer included in TEX Live as a separate program; only Aleph is provided:

Aleph combines the Ω and ε-TEX extensions. See `texmf-dist/doc/aleph/base`.

2.5 Other notable programs in TEX Live

Here are a few other commonly-used programs included in TEX Live:

bibtex bibliography support.

makeindex, xindy index support.

dvips convert DVI to PostScript.

xdvi DVI previewer for the X Window System.

dviconcat, dviselect cut and paste pages from DVI files.

dvipdfmx convert DVI to PDF, an alternative approach to pdfTEX (mentioned above).

psselect, psnup, ... PostScript utilities.

pdfjam, pdfjoin, ... PDF utilities.

context, mtxrun ConTEXt and PDF processor.

htlatex, ... tex4ht: (LA)TEX to HTML (and XML and more) converter.

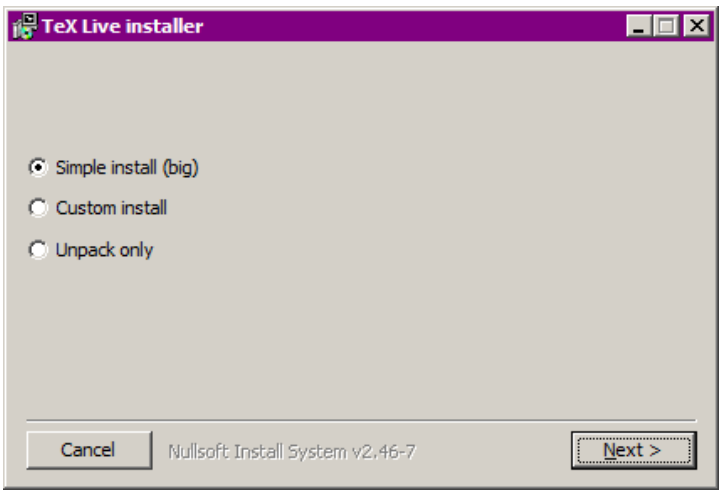


Figure 1: First stage of Windows .exe installer

3 Installation

3.1 Starting the installer

To begin, get the TEX Collection DVD or download the TEX Live net installer. See http://tug.org/texlive/acquire.html for more information and other methods of getting the software.

Net installer, .zip or .tar.gz: Download from CTAN, under `systems/texlive/tlnet`; the url http://mirror.ctan.org/systems/texlive/tlnet should redirect to a nearby, up-to-date, mirror. You can retrieve either `install-tl.zip` which can be used under Unix and Windows, or the considerably smaller `install-unx.tar.gz` for Unix only. After unpacking, `install-tl` and `install-tl-windows.bat` will be in the `install-tl` subdirectory.

Net installer, Windows .exe: Download from CTAN as above, and double-click. This starts up a first-stage installer and unpacker; see Figure 1. It gives three choices: "Simple install" starts up the wizard installer, "Custom install" the expert GUI installer, as described in section 3.1.3. A third option is just unpacking.

TEX Collection DVD: go to the DVD's `texlive` subdirectory. Under Windows, the installer normally starts automatically when you insert the DVD. You can get the DVD by becoming a member of a TEX user group (highly recommended, http://tug.org/usergroups.html), or purchasing it separately (http://tug.org/store), or burning your own from the ISO image. You can also mount the ISO directly on most systems. After installing from DVD or ISO, if you want to get continuing updates from the Internet, please see 3.4.3.

The same installer program is run, whatever the source. The most visible difference between the two is that with the net installer, what you end up with is the packages that are currently available. This is in contrast to the DVD and ISO images, which are not updated between the major public releases.

If you need to download through proxies, use a `~/.wgetrc` file or environment variables with the proxy settings for Wget (http://www.gnu.org/software/wget/manual/html_node/Proxies.html). TEX Live always uses GNU Wget to download. Of course, this should not matter if you are installing from the DVD or ISO image.
The following sections explain installer start-up in more detail.

3.1.1 Unix

Below, `>` denotes the shell prompt; user input is **`bold`**. The script `install-tl` is a Perl script. The simplest way to start it on a Unix-compatible system is as follows:

```
> perl /path/to/installer/install-tl
```

(Or you can invoke **`/path/to/installer/install-tl`** if it stayed executable, or `cd` to the directory first, etc.; we won't repeat all the variations.) You may have to enlarge your terminal window so that it shows the full text installer screen (Figure 2).

To install in expert GUI mode (figure 3), you'll need the `Perl::TK` module compiled with XFT support, which is generally the case with GNU/Linux, but often not so with other systems. Given that, you can run:

```
> perl install-tl -gui
```

For a complete listing of the various options:

```
> perl install-tl -help
```

Warning about Unix permissions: Your `umask` at the time of installation will be respected by the TEX Live installer. Therefore, if you want your installation to be usable by users other than you, make sure your setting is sufficiently permissive, for instance, `umask 002`. For more information about `umask`, consult your system documentation.

Special considerations for Cygwin: Unlike other Unix-compatible systems, Cygwin does not by default include all of the prerequisite programs needed by the TEX Live installer. See Section 3.1.4.

3.1.2 Mac OS X

As mentioned in section 2.1, a separate distribution is prepared for Mac OS X, named MacTEX (`http://tug.org/mactex`). We recommend using the native MacTEX installer instead of the TEX Live installer on Mac OS X, because the native installer makes a few Mac-specific adjustments, in particular to allow easily switching between the various TEX distributions for Mac OS X (MacTEX, Fink, MacPorts, ...) using the so-called TEXDist data structure.

MacTEX is firmly based on TEX Live, and the main TEX trees and binaries are precisely the same. It adds a few extra folders with Mac-specific documentation and applications.

3.1.3 Windows

If you are using the unpacked downloaded zip file, or the DVD installer failed to start automatically, double-click `install-tl-windows.bat`. If you want more customization options, e.g., selection of specific package collections, run `install-tl-advanced.bat` instead.

You can also start the installer from the command-prompt. Below, `>` denotes the prompt; user input is **bold**. If you are in the installer directory, run just:

```
> install-tl-windows
```

Or you can invoke it with an absolute location, such as:

```
> D:\texlive\install-tl-windows
```

for the TEX Collection DVD, supposing that `D:` is the optical drive. Figure 4 displays the wizard installer, which is the default for Windows.

To install in text mode, use:

```
> install-tl-windows -no-gui
```

For a complete listing of the various options:

```
> install-tl-windows -help
```

3.1.4 Cygwin

Before beginning the installation, use Cygwin's `setup.exe` program to install the `perl` and `wget` packages if you have not already done so. The following additional packages are recommended:

- `fontconfig` [needed by XeTEX and LuaTEX]
- `ghostscript` [needed by various utilities]
- `libXaw7` [needed by `xdvi`]
- `ncurses` [provides the `clear` command used by the installer]

```
Installing TeX Live 2015 from: ...
Platform: i386-linux => 'GNU/Linux on Intel x86'
Distribution: inst (compressed)
Directory for temporary files: /tmp
...
 Detected platform: GNU/Linux on Intel x86

 <B> binary platforms: 1 out of 19

 <S> set installation scheme (scheme-full)

 <C> customizing installation collections
     47 collections out of 48, disk space required: 4022 MB

 <D> directories:
   TEXDIR (the main TeX directory):
     /usr/local/texlive/2015
   ...

 <O> options:
   [ ] use letter size instead of A4 by default
   ...

 <V> set up for portable installation

Actions:
 <I> start installation to hard disk
 <H> help
 <Q> quit
```

Figure 2: Main text installer screen (GNU/Linux)

3.1.5 The text installer

Figure 2 displays the main text mode screen under Unix. The text installer is the default on Unix.

This is only a command-line installer; there is no cursor support at all. For instance, you cannot tab around checkboxes or input fields. You just type something (case-sensitive) at the prompt and press the Enter key, and then the entire terminal screen will be rewritten, with adjusted content.

The text installer interface is this primitive for a reason: it is designed to run on as many platforms as possible, even with a very barebones Perl.

3.1.6 The expert graphical installer

Figure 3 displays the expert graphical installer under GNU/Linux. Other than using buttons and menus, this does not differ much from the text installer.

This mode can be invoked explicitly with

```
> install-tl -gui=perltk
```

3.1.7 The simple wizard installer

Under Windows, the default is to run the simplest installation method we could devise, called the "wizard" installer (figure 4). It installs everything and asks almost no questions. If you want to customize your setup, you should run one of the other installers.

On other platforms, this mode can be invoked explicitly with

```
> install-tl -gui=wizard
```

3.2 Running the installer

The installer is intended to be mostly self-explanatory, but following are a few notes about the various options and submenus.

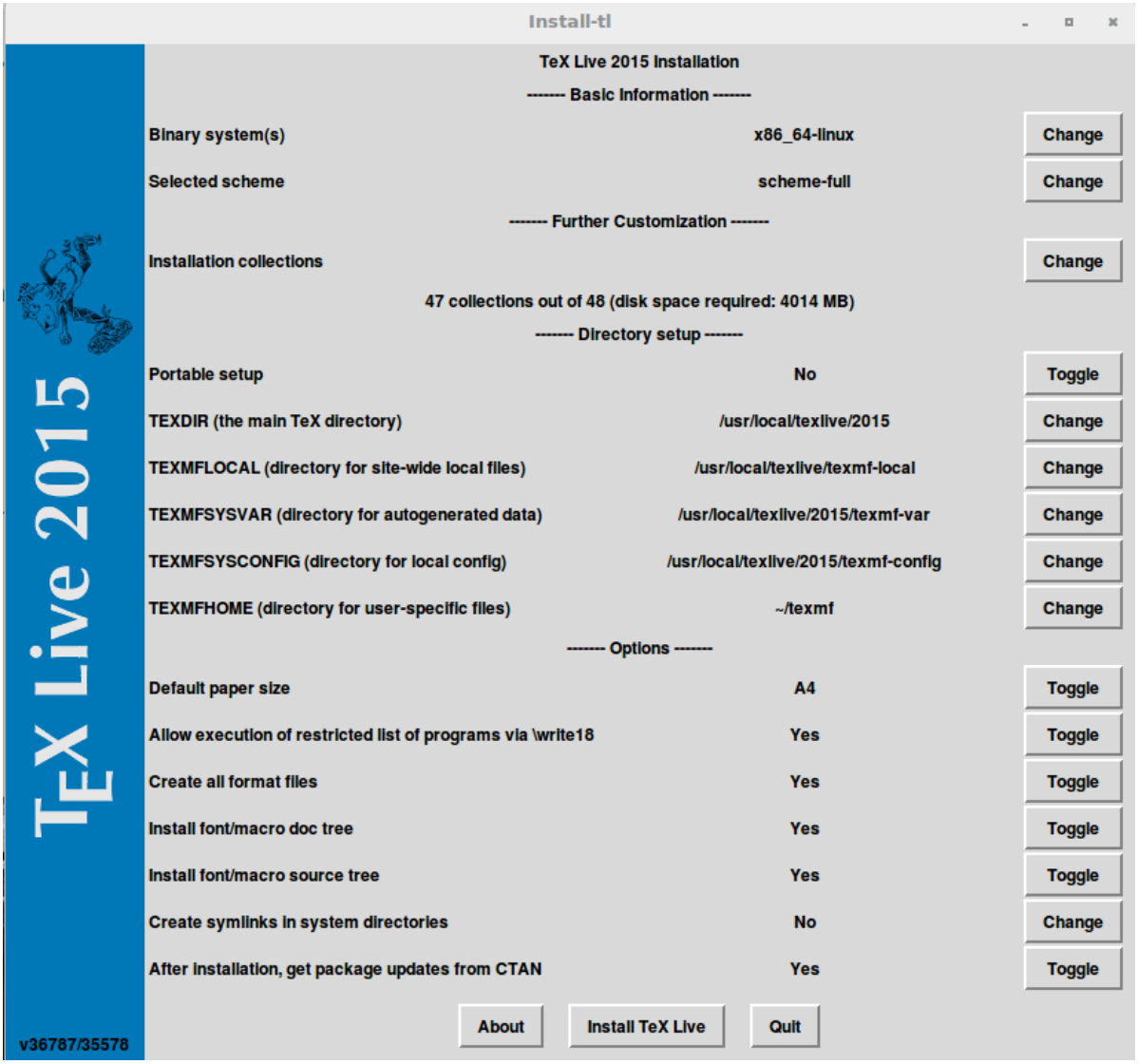


Figure 3: Expert GUI installer screen (GNU/Linux)

3.2.1 Binary systems menu (Unix only)

Figure 5 displays the text mode binaries menu. By default, only the binaries for your current platform will be installed. From this menu, you can select installation of binaries for other platforms as well. This can be useful if you are sharing a TEX tree across a network of heterogeneous machines, or for a dual-boot system.

3.2.2 Selecting what is to be installed

Figure 6 displays the TEX Live scheme menu; from here, you choose a "scheme", which is an overall set of package collections. The default `full` scheme installs everything available. This is recommended, but you can also choose the `basic` scheme for a small system, `minimal` for testing purposes, and `medium` or `teTeX` to get something in between. There are also various specialized and country-specific schemes.

You can refine your scheme selection with the 'collections' menu (figure 7, shown in GUI mode for a change).

Collections are one level more detailed than schemes — in essence, a scheme consists of several collections, a collection consists of one or more packages, and a package (the lowest level grouping in TEX Live) contains the actual TEX macro files, font files, and so on.

If you want more control than the collection menus provide, you can use the TEX Live Manager (tlmgr) program after installation (see section 5); using that, you can control the installation at the package level.

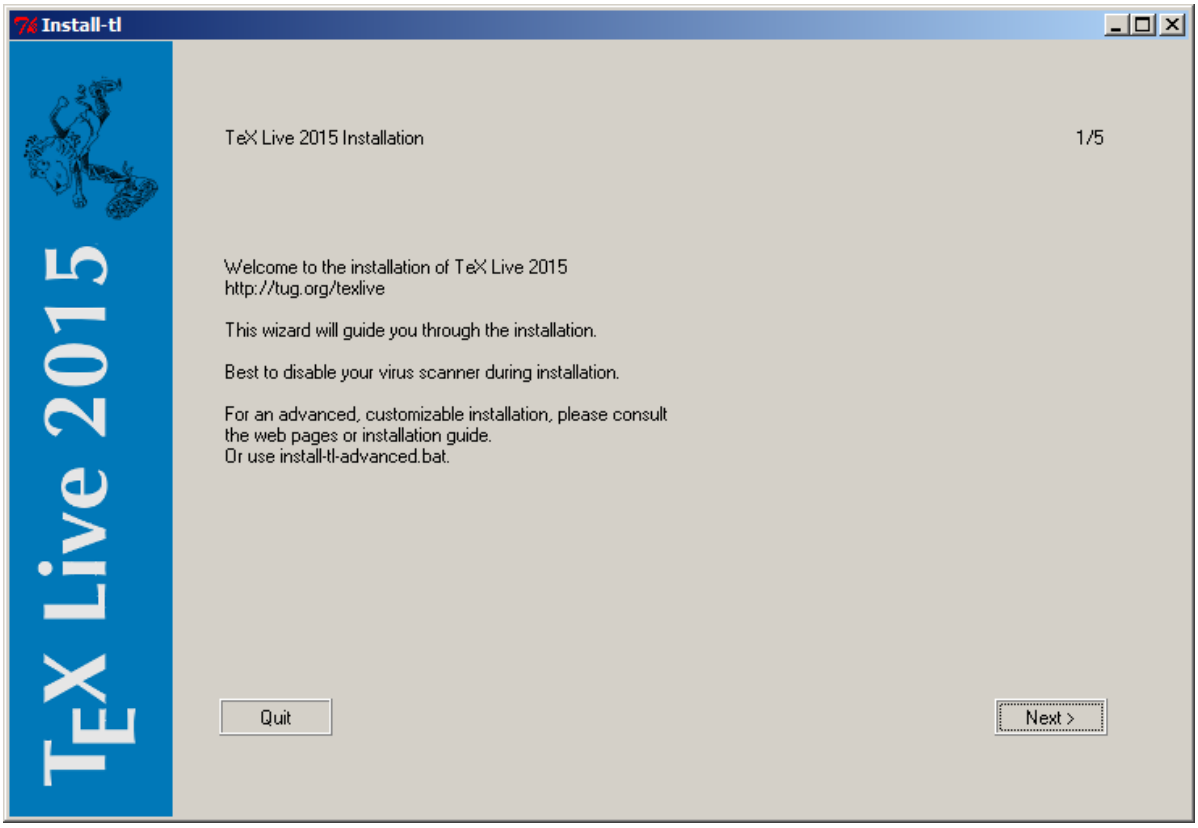


Figure 4: Wizard installer screen (Windows)

```
Available platforms:
================================================================================
   a [ ] Cygwin on Intel x86 (i386-cygwin)
   b [ ] Cygwin on x86_64 (x86_64-cygwin)
   c [ ] MacOSX/Darwin universal binaries (universal-darwin)
   d [ ] MacOSX/Darwin on x86_64 (x86_64-darwin)
   e [ ] FreeBSD on x86_64 (amd64-freebsd)
   f [ ] FreeBSD on Intel x86 (i386-freebsd)
   g [ ] GNU/Linux on DEC Alpha (alpha-linux)
   h [ ] GNU/Linux on ARM (armel-linux)
   i [ ] GNU/Linux on ARMhf (armhf-linux)
   j [X] GNU/Linux on Intel x86 (i386-linux)
   k [ ] GNU/Linux on MIPSel (mipsel-linux)
   l [ ] GNU/Linux on PowerPC (powerpc-linux)
   m [ ] GNU/Linux on x86_64 (x86_64-linux)
   o [ ] NetBSD on x86_64 (amd64-netbsd)
   p [ ] NetBSD on Intel x86 (i386-netbsd)
   s [ ] Solaris on Intel x86 (i386-solaris)
   t [ ] Solaris on Sparc (sparc-solaris)
   u [ ] Solaris on x86_64 (x86_64-solaris)
   v [ ] Windows (win32)
```

Figure 5: Binaries menu

3.2.3 Directories

The default layout is described in section 2.3, p. 5. The default installation directory is `/usr/local/texlive/2015` on Unix and `%SystemDrive%\texlive\2015` on Windows. This arrangement enables having many parallel TEX Live installations, such as one for each release (typically by year, as here), and you can switch between them merely by altering your search path.

That installation directory can be overridden by setting the so-called `TEXDIR` in the installer. The GUI screen for this and other options is shown in figure 3. The most common reasons to change it are either lacking enough disk space in that partition (the full TEX Live needs several gigabytes), or lacking write permission for the default location (you don't have to be root or administrator to install TEX

```
Select scheme:
================================================================================
 a [X] full scheme (everything)
 b [ ] medium scheme (small + more packages and languages)
 c [ ] small scheme (basic + xetex, metapost, a few languages)
 d [ ] basic scheme (plain and latex)
 e [ ] minimal scheme (plain only)
 f [ ] ConTeXt scheme
 g [ ] GUST TeX Live scheme
 h [ ] teTeX scheme (more than medium, but nowhere near full)
 i [ ] XML scheme
 j [ ] custom selection of collections
```

Figure 6: Scheme menu

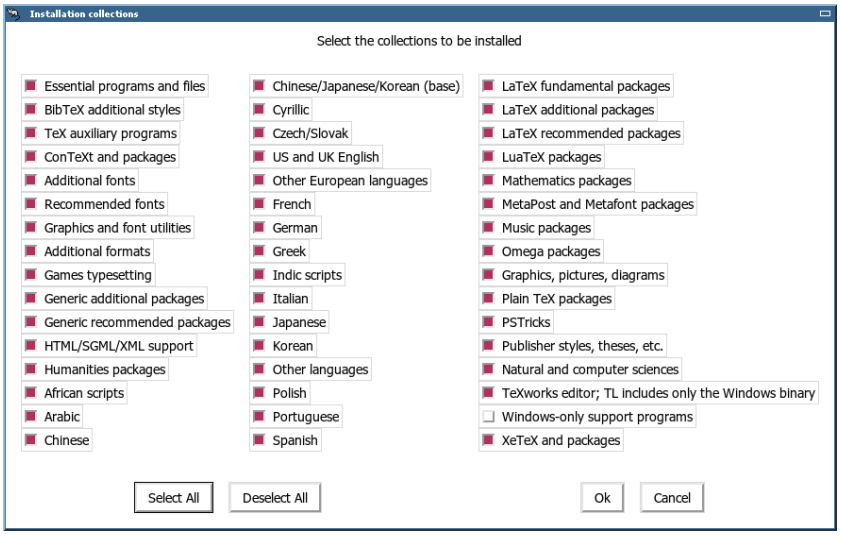


Figure 7: Collections menu

Live, but you do need write access to the target directory).

The installation directories can also be changed by setting a variety of environment variables before running the installer (most likely, `TEXLIVE_INSTALL_PREFIX` or `TEXLIVE_INSTALL_TEXDIR`); see the documentation from `install-tl --help` (available online at http://tug.org/texlive/doc/install-tl.html) for the full list and more details.

A reasonable alternative destination is a directory under your home, especially if you will be the sole user. Use '`~`' to indicate this, as in '`~/texlive/2015`'.

We recommend including the year in the name, to enable keeping different releases of TEX Live side by side. (You may wish to also maintain a version-independent name, such as `/usr/local/texlive-cur`, via a symbolic link, which you can then repoint after testing the new release.)

Changing `TEXDIR` in the installer will also change `TEXMFLOCAL`, `TEXMFSYSVAR` and `TEXMFSYSCONFIG`.

`TEXMFHOME` is the recommended location for personal macro files or packages. The default value is `~/texmf`. In contrast to `TEXDIR`, here a `~` is preserved in the newly-written configuration files, since it usefully refers to the home directory of the user running TEX. It expands to `$HOME` on Unix and `%USERPROFILE%` on Windows. Special redundant note: `TEXMFHOME`, like all trees, must be organized according to the TDS, or files may not be found.

`TEXMFVAR` is the location for storing most cached runtime data specific to each user. `TEXMFCACHE` is the variable name used for that purpose by LuaLATEXand ConTEXt MkIV (see section 3.4.5, p. 15); its default value is `TEXMFSYSVAR`, or (if that's not writable), `TEXMFVAR`.

3.2.4 Options

Figure 8 shows the text mode options menu. More info on each:

use letter size instead of A4 by default: The default paper size selection. Of course, individual documents can and should specify a specific paper size, if desired.

```
Options setup:
================================================================================
 <P> use letter size instead of A4 by default: [ ]
 <E> execution of restricted list of programs: [X]
 <F> create format files:                      [X]
 <D> install font/macro doc tree:              [X]
 <S> install font/macro source tree:           [X]
 <L> create symlinks in standard directories:  [ ]
            binaries to:
            manpages to:
                info to:
 <Y> after installation, get package updates from CTAN: [X]
```

Figure 8: Options menu (Unix)

execution of restricted list of programs: As of TEX Live 2010, execution of a few external programs is allowed by default. The (very short) list of allowed programs is given in the `texmf.cnf`. See the 2010 news (section 9.1.7) for more details.

create format files: Although unnecessary format files take time to generate and disk space to store, it is still recommended to leave this option checked: if you don't, then format files will be generated in people's private `TEXMFVAR` tree as they are needed. In that location, they will not be updated automatically if (say) binaries or hyphenation patterns are updated in the installation, and thus you could end up with incompatible format files.

install font/macro ... tree: Omit downloading/installing the documentation and source files included in most packages. Not recommended.

create symlinks in standard directories: This option (Unix only) bypasses the need to change environment variables. Without this option, TEX Live directories usually have to be added to `PATH`, `MANPATH` and `INFOPATH`. You will need write permissions to the target directories. It is strongly advised *not* to overwrite a TEX system that came with your system with this option. It is intended primarily for accessing the TEX system through directories that are already known to users, such as `/usr/local/bin`, which don't already contain any TEX files.

after installation ... CTAN: When installing from DVD, this option is enabled by default, since usually one wants to take any subsequent package updates from the CTAN area that is updated throughout the year. The only likely reason to disable it is if you install only a subset from the DVD and plan to augment the installation later. In any case, the package repository for the installer, and for updates after installation, can be set independently as needed; see section 3.3.1 and section 3.4.3.

When all the settings are to your liking, you can type 'I' to start the installation process. When it is done, skip to section 3.4 to read what else needs to be done, if anything.

3.3 Command-line install-tl options

Type

```
> install-tl -help
```

for a listing of command-line options. Either `-` or `--` can be used to introduce option names. These are the most common ones:

`-gui` If possible, use the GUI installer. This requires the Perl/Tk module (http://tug.org/texlive/distro.html#perltk) compiled with XFT support; if Perl/Tk is not available, installation continues in text mode.

`-no-gui` Force using the text mode installer, even under Windows.

`-lang` *LL* Specify the installer interface language as a standard (usually two-letter) code. The installer tries to automatically determine the right language but if it fails, or if the right language is not available, then it uses English as a fallback. Run `install-tl --help` to get the list of available languages.

`-portable` Install for portable use on, e.g., a USB stick. Also selectable from within the text installer with the `V` command, and from the GUI installer. See section 4.2.

`-profile` *file* Load the installation profile *file* and do the installation with no user interaction. The installer always writes a file `texlive.profile` to the `tlpkg` subdirectory of your installation. That file can be given as the argument to redo the exact same installation on a different system, for example. Alternatively, you can use a custom profile, most easily created by starting from a generated one and changing values, or an empty file, which will take all the defaults.

`-repository` *url-or-directory* Specify package repository from which to install; see following.

`-in-place` (Documented only for completeness: Do not use this unless you know what you are doing.) If you already have an rsync, svn, or other copy of TeX Live (see `http://tug.org/texlive/acquire-mirror.html`) then this option will use what you've got, as-is, and do only the necessary post-install. Be warned that the file `tlpkg/texlive.tlpdb` may be overwritten; saving it is your responsibility. Also, package removal has to be done manually. This option cannot be toggled via the installer interface.

3.3.1 The `-repository` option

The default network package repository is a CTAN mirror chosen automatically via `http://mirror.ctan.org`.

If you want to override that, the location value can be a url starting with `ftp:`, `http:`, or `file:/`, or a plain directory path. (When giving an `http:` or `ftp:` location, trailing '`/`' characters and/or a trailing '`/tlpkg`' component are ignored.)

For example, you could choose a particular CTAN mirror with something like: `http://ctan.example.org/tex-archive/systems/texlive/tlnet/`, substituting a real hostname and its particular top-level CTAN path for `ctan.example.org/tex-archive`. The list of CTAN mirrors is maintained at `http://ctan.org/mirrors`.

If the given argument is local (either a path or a `file:/` url), compressed files in an `archive` subdirectory of the repository path are used (even if uncompressed files are available as well).

3.4 Post-install actions

Some post-installation may be required.

3.4.1 Environment variables for Unix

If you elected to create symlinks in standard directories (described in section 3.2.4), then there is no need to edit environment variables. Otherwise, on Unix systems, the directory of the binaries for your platform must be added to the search path. (On Windows, the installer takes care of this.)

Each supported platform has its own subdirectory under `TEXDIR/bin`. See figure 5 for the list of subdirectories and corresponding platforms.

Optionally, you can also add the documentation man and Info directories to their respective search paths, if you want the system tools to find them. The man pages might be found automatically after the addition to `PATH`.

For Bourne-compatible shells such as `bash`, and using Intel x86 GNU/Linux and a default directory setup as an example, the file to edit might be `$HOME/.profile` (or another file sourced by `.profile`, and the lines to add would look like this:

```
PATH=/usr/local/texlive/2015/bin/i386-linux:$PATH; export PATH
MANPATH=/usr/local/texlive/2015/texmf-dist/doc/man:$MANPATH; export MANPATH
INFOPATH=/usr/local/texlive/2015/texmf-dist/doc/info:$INFOPATH; export INFOPATH
```

For csh or tcsh, the file to edit is typically `$HOME/.cshrc`, and the lines to add might look like:

```
setenv PATH /usr/local/texlive/2015/bin/i386-linux:$PATH
setenv MANPATH /usr/local/texlive/2015/texmf-dist/doc/man:$MANPATH
setenv INFOPATH /usr/local/texlive/2015/texmf-dist/doc/info:$INFOPATH
```

If you already have settings somewhere in your "dot" files, naturally the TeX Live directories should be merged in as appropriate.

3.4.2 Environment variables: Global configuration

If you want to make these changes globally, or for a user newly added to the system, then you are on your own; there is just too much variation between systems in how and where these things are configured.

Our two hints are: 1) you may want to check for a file `/etc/manpath.config` and, if present, add lines such as

```
MANPATH_MAP /usr/local/texlive/2015/bin/i386-linux \
            /usr/local/texlive/2015/texmf-dist/doc/man
```

And 2) check for a file `/etc/environment` which may define the search path and other default environment variables.

In each (Unix) binary directory, we also create a symbolic link named `man` to the directory `texmf-dist/doc/man`. Some `man` programs, such as the standard Mac OS X `man`, will automatically find that, obviating the need for any man page setup.

3.4.3 Internet updates after DVD installation

If you installed TEX Live from DVD and then wish to get updates from the Internet, you need to run this command—*after* you've updated your search path (as described in the previous section):

```
> tlmgr option repository http://mirror.ctan.org/systems/texlive/tlnet
```

This tells tlmgr to use a nearby CTAN mirror for future updates. This is done by default when installing from DVD, via the option described in section 3.2.4.

If there are problems with the automatic mirror selection, you can specify a particular CTAN mirror from the list at http://ctan.org/mirrors. Use the exact path to the `tlnet` subdir on that mirror, as shown above.

3.4.4 System font configuration for XeTEX and LuaTEX

XeTEX and LuaTEX can use any font installed on the system, not just those in the TEX trees. They do these via related but not identical methods.

On Windows, fonts shipped with TEX Live are made available to XeTEX automatically. But if you have installed the `xetex` package on a Unix-compatible system, you need to configure your system to be able to find the fonts shipped with TEX Live via system name lookup, and not just filename lookup.

To facilitate this, when the xetex package is installed (either at initial installation or later), the necessary configuration file is created in `TEXMFSYSVAR/fonts/conf/texlive-fontconfig.conf`.

To set up the TEX Live fonts for system-wide use (assuming you have suitable privileges), proceed as follows:

1. Copy the `texlive-fontconfig.conf` file to `/etc/fonts/conf.d/09-texlive.conf`.
2. Run `fc-cache -fsv`.

If you do not have sufficient privileges to carry out the steps above, or if you want to make the TEX Live fonts available to only one user, you can do the following:

1. Copy the `texlive-fontconfig.conf` file to `~/.fonts.conf`, where `~` is your home directory.
2. Run `fc-cache -fv`.

You can run `fc-list` to see the names of the system fonts. The incantation `fc-list : family style file spacing` (all arguments are literal strings) shows some generally interesting information.

3.4.5 ConTEXt Mark IV

Both the 'old' ConTEXt (Mark II) and the 'new' ConTEXt (Mark IV) should run out of the box after TEX Live installation, and should need no special attention as long as you stick to using `tlmgr` for updates.

However, because ConTEXt MkIV does not use the kpathsea library, some setup will be required whenever you install new files manually (without using `tlmgr`). After each such installation, each MkIV user must run:

```
context --generate
```

to refresh the ConTEXt disk cache data. The resulting files are stored under `TEXMFCACHE`. whose default value in TEX Live is `TEXMSYSVAR;TEXMFVAR`.

ConTEXt MkIV will read from all paths mentioned in `TEXMFCACHE`, and write to the first path that is writable. While reading, the last found match will take precedence in the case of duplicated cache data.

For more information, see http://wiki.contextgarden.net/Running_Mark_IV.

3.4.6 Integrating local and personal macros

This is already mentioned implicitly in section 2.3: `TEXMFLOCAL` (by default, `/usr/local/texlive/texmf-local` or `%SystemDrive%\texlive\texmf-local` on Windows) is intended for system-wide local fonts and macros; and `TEXMFHOME` (by default, `$HOME/texmf` or `%USERPROFILE%\texmf`), is for personal fonts and macros. These directories are intended to stick around from release to release, and have their content seen automatically by a new TEX Live release. Therefore, it is best to refrain from changing the definition of `TEXMFLOCAL` to be too far away from the main TEX Live directory, or you will need to manually change future releases.

For both trees, files should be placed in their proper TEX Directory Structure (TDS) subdirectories; see http://tug.org/tds or consult `texmf-dist/web2c/texmf.cnf`. For instance, a LATEX class file or package should be placed in `TEXMFLOCAL/tex/latex` or `TEXMFHOME/tex/latex`, or a subdirectory thereof.

`TEXMFLOCAL` requires an up-to-date filename database, or files will not be found. You can update it with the command mktexlsr or use the 'Reinit file database' button on the configuration tab of the TEX Live Manager GUI.

By default, each of these variables is defined to be a single directory, as shown. This is not a hard-and-fast requirement. If you need to easily switch back and forth between different versions of large packages, for example, you can maintain multiple trees for your own purposes. This is done by setting `TEXMFHOME` to the list of directories, within braces, separated by commas:

```
TEXMFHOME = {/my/dir1,/mydir2,/a/third/dir}
```

Section 7.1.5 describes brace expansion further.

3.4.7 Integrating third-party fonts

This is unfortunately a messy topic. Forget about it unless you want to delve into many details of the TEX installation. Many fonts are included in TEX Live already, so take a look if you don't already know that what you want isn't there.

A possible alternative is to use XeTEX or LuaTEX (see section 2.4), which let you use operating system fonts without any installation in TEX.

If you do need to do this, see http://tug.org/fonts/fontinstall.html for our best effort at describing the procedure.

3.5 Testing the installation

After installing TEX Live, you naturally want to test it out, so you can start creating beautiful documents and/or fonts.

One thing you may immediately be looking for is a front-end with which to edit files. TEX Live installs TEXworks (http://tug.org/texworks) on Windows (only), and MacTEX installs TeXShop (http://pages.uoregon.edu/koch/texshop. On other Unix systems, it's left up to you to choose an editor. There are many choices available, some of which are listed in the next section; see also http://tug.org/interest.html#editors. Any plain text editor will work; something TEX-specific is not required.

The rest of this section gives some basic procedures for testing that the new system is functional. We give Unix commands here; under Mac OS X and Windows, you're more likely to run the tests through a graphical interface, but the principles are the same.

1. Make sure that you can run the tex program in the first place:

```
> tex --version
TeX 3.14159265 (TeX Live ...)
Copyright ... D.E. Knuth.
...
```

If this comes back with 'command not found' instead of version and copyright information, or with an older version, most likely you don't have the correct `bin` subdirectory in your `PATH`. See the environment-setting information on p. 14.

2. Process a basic LaTeX file:

```
> latex sample2e.tex
This is pdfTeX 3.14...
...
Output written on sample2e.dvi (3 pages, 7484 bytes).
Transcript written on sample2e.log.
```

If this fails to find `sample2e.tex` or other files, most likely you have interference from old environment variables or configuration files; we recommend unsetting all TeX-related environment variables for a start. (For a deep analysis, you can ask TeX to report on exactly what it is searching for, and finding; see "Debugging actions" on page 30.)

3. Preview the result online:

```
> xdvi sample2e.dvi    # Unix
> dviout sample2e.dvi  # Windows
```

You should see a new window with a nice document explaining some of the basics of LaTeX. (Well worth reading, by the way, if you're new to TeX.) You do have to be running under X for `xdvi` to work; if you're not, or your `DISPLAY` environment variable is set incorrectly, you'll get an error '`Can't open display`'.

4. Create a PostScript file for printing or display:

```
> dvips sample2e.dvi -o sample2e.ps
```

5. Create a PDF file instead of DVI; this processes the `.tex` file and writes PDF directly:

```
> pdflatex sample2e.tex
```

6. Preview the PDF file:

```
> gv sample2e.pdf
```
or:
```
> xpdf sample2e.pdf
```

Neither `gv` nor `xpdf` are included in TeX Live, so you must install them separately. See http://www.gnu.org/software/gv and http://www.foolabs.com/xpdf, respectively. There are plenty of other PDF viewers, too. For Windows, we recommend trying Sumatra PDF (http://blog.kowalczyk.info/software/sumatrapdf).

7. Standard test files you may find useful in addition to `sample2e.tex`:

`small2e.tex` A simpler document than `sample2e`, to reduce the input size if you're having troubles.

`testpage.tex` Test if your printer introduces any offsets.

`nfssfont.tex` For printing font tables and tests.

`testfont.tex` Also for font tables, but using plain TeX.

`story.tex` The most canonical (plain) TeX test file of all. You must type '`\bye`' to the `*` prompt after '`tex story.tex`'.

8. If you have installed the `xetex` package, you can test its access to system fonts as follows:

```
> xetex opentype-info.tex
This is XeTeX, Version 3.14...
...
Output written on opentype-info.pdf (1 page).
Transcript written on opentype-info.log.
```

If you get an error message saying "Invalid fontname 'Latin Modern Roman/ICU'...", then you need to configure your system so that the fonts shipped with TeX Live can be found. See Section 3.4.4.

3.6 Links for additional downloadable software

If you are new to TEX, or otherwise need help with actually writing TEX or LATEX documents, please visit http://tug.org/begin.html for some introductory resources.

Links for some other tools you may consider installing:

Ghostscript http://www.cs.wisc.edu/~ghost/

Perl http://www.perl.org/ with supplementary packages from CPAN, http://www.cpan.org/

ImageMagick http://www.imagemagick.com, for graphics processing and conversion

NetPBM http://netpbm.sourceforge.net/, also for graphics.

TEX-oriented editors There is a wide choice, and it is a matter of the user's taste. Here is a selection in alphabetical order (a few here are for Windows only).

- GNU Emacs is available natively under Windows, see http://www.gnu.org/software/emacs/windows/ntemacs.html.
- Emacs with AucTEX for Windows is available from CTAN. The AuCTEX home page is http://www.gnu.org/software/auctex.
- LEd is available from http://mirror.ctan.org/support/LEd.
- SciTE is available from http://www.scintilla.org/SciTE.html.
- Texmaker is free software, available from http://www.xm1math.net/texmaker.
- TeXstudio started out as a fork of Texmaker with additional features; http://texstudio.sourceforge.net.
- TeXnicCenter is free software, available from http://www.texniccenter.org and in the proTEXt distribution.
- TeXworks is free software, available from http://tug.org/texworks and installed as part of TEX Live for Windows (only).
- Vim is free software, available from http://www.vim.org.
- WinEdt is shareware available though http://tug.org/winedt or http://www.winedt.com.
- WinShell is available from http://www.winshell.de.

For a much longer list of packages and programs, see http://tug.org/interest.html.

4 Specialized installations

The previous sections described the basic installation process. Here we turn to some specialized cases.

4.1 Shared-user (or cross-machine) installations

TEX Live has been designed to be sharable between different users on one system, and/or between different systems on a network. With a standard directory layout, no hard paths are configured: the locations for files needed by TEX Live programs are found relative to the programs. You can see this in the principal configuration file `$TEXMFDIST/web2c/texmf.cnf`, which contains lines such as

```
TEXMFROOT = $SELFAUTOPARENT
...
TEXMFDIST = $TEXMFROOT/texmf-dist
...
TEXMFLOCAL = $SELFAUTOPARENT/../texmf-local
```

This means that adding the directory for TEX Live executables for their platform to their search path is sufficient to get a working setup.

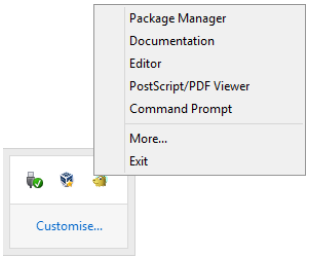
By the same token, you can also install TEX Live locally and then move the entire hierarchy afterwards to a network location.

For Windows, a sample network installation script named `w32client` can be downloaded through http://tug.org/texlive/w32client.html. It creates settings and menu shortcuts for using an existing TEX Live installation on a LAN. It also registers an uninstaller `w32unclient`, available in the same zip file. See the web page for more information.

4.2 Portable (USB) installations

The `-portable` installer option (or `V` command in the text installer or corresponding GUI option) creates a completely self-contained TEX Live installation under a common root and forgoes system integration. You can create such an installation directly on a USB stick, or copy it to a USB stick afterwards.

To run TEX using this portable installation, you need to add the appropriate binary directory to the search path during your terminal session, as usual. On Windows, you can double-click `tl-tray-menu` at the root of the installation to choose between a few common tasks, as shown in this screenshot:



The 'More...' entry explains how you can customize this menu.

4.3 ISO (or DVD) installations

If you don't need to update or otherwise modify your installation often, and/or have several systems on which to use TEX Live, it may be convenient to create an ISO of your TEX Live installation, because:

- Copying an ISO between different computers is much faster than copying a normal installation.
- If you are dual-booting between different operating systems and want them to share a TEX Live installation, an ISO installation is not tied to the idiosyncrasies and limitations of other mutually supported filesystems (FAT32, NTFS, HFS+).
- Virtual machines can simply mount such an ISO.

Of course you can also burn an ISO to DVD, if that is useful for you.

Desktop GNU/Linux/Unix systems, including Mac OS X, are able to mount an ISO. Windows 8 is the first(!) Windows version which can do this. Apart from that, nothing changes compared to a normal hard disk installation, see section 3.4.1.

When preparing such an ISO installation, it is best to omit the subdirectory for the release year, and have `texmf-local` at the same level as the other trees (`texmf-dist`, `texmf-var`, etc.). You can do this with the normal directory options in the installer.

For a physical (rather than virtual) Windows system, you can burn the ISO to DVD. However, it may be worth your while to investigate free ISO-mounting options such as WinCDEmu at `http://wincdemu.sysprogs.org/`.

For Windows system integration, you can include the `w32client` scripts described in section 4.1 and at `http://tug.org/texlive/w32client.html`, which work just as well for an ISO as for a network installation.

On Mac OS X, TeXShop will be able to use the DVD installation if a symlink `/usr/texbin` points to the appropriate binary directory, e.g.,

```
sudo ln -s /Volumes/MyTeXLive/bin/universal-darwin /usr/texbin
```

Historical note: TEX Live 2010 was the first TEX Live edition which was no longer distributed 'live'. However, it always required some acrobatics to run from DVD or ISO; in particular, there was no way around setting at least one extra environment variable. If you create your ISO from an existing installation then there is no need for this.

5 tlmgr: Managing your installation

TEX Live includes a program named tlmgr for managing TEX Live after the initial installation. Its capabilities include:

- installing, updating, backing up, restoring, and uninstalling individual packages, optionally taking dependencies into account;

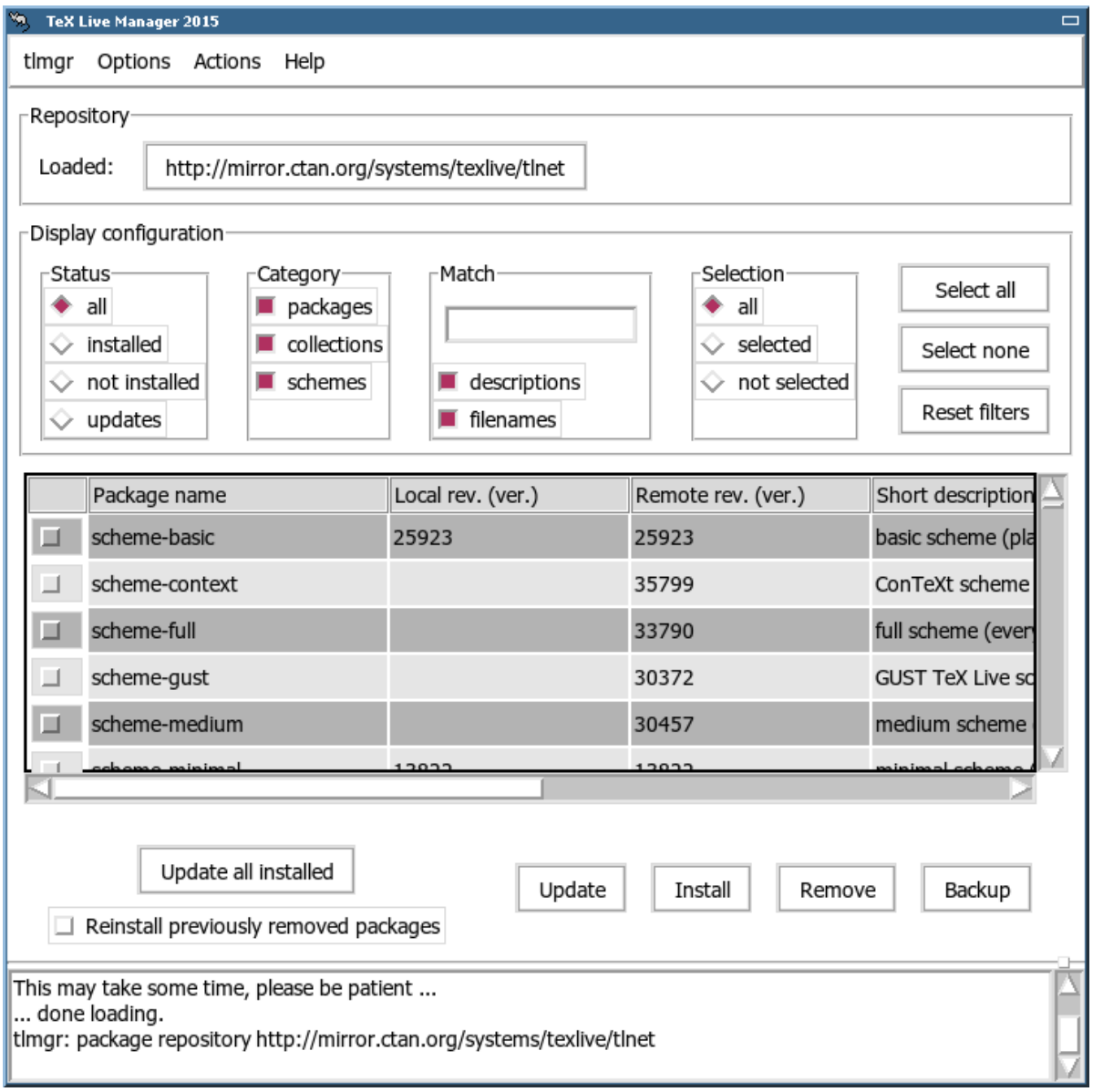


Figure 9: tlmgr in GUI mode: main window, after 'Load'.

- searching for and listing packages and their descriptions;
- listing, adding, and removing platforms;
- changing installation options such as paper size and source location (see section 3.3.1).

tlmgr's functionality completely subsumes the texconfig program. We still distribute and maintain texconfig for the sake of anyone used to its interface, but we recommend using tlmgr nowadays.

5.1 tlmgr GUI mode

tlmgr can be started in GUI mode (figure 9) with

```
> tlmgr -gui
```

or in Windows via the Start menu: `Start`, `Programs`, `TeX Live ...`, `TeX Live Manager`. After clicking 'Load' it displays a list of available and installed packages. This assumes of course that the installation source is valid and reachable.

Figures 10 and 11 show the general and paper size option screens.

5.2 Sample tlmgr command-line invocations

After the initial installation, you can update your system to the latest versions available with:

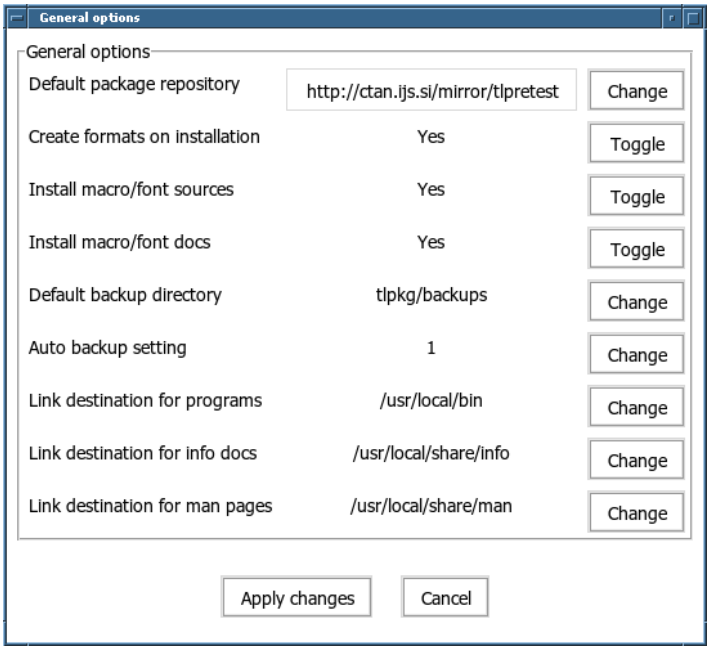


Figure 10: `tlmgr` in GUI mode: General options

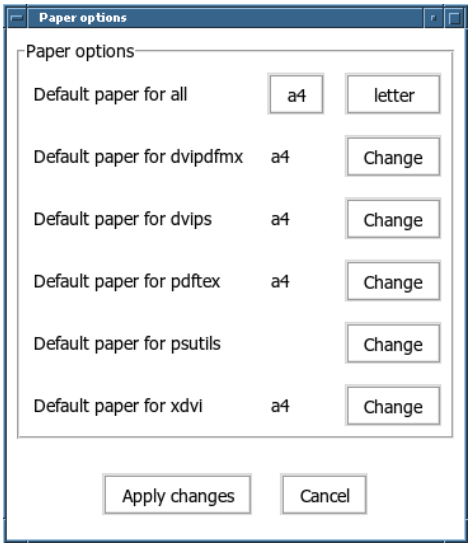


Figure 11: `tlmgr` in GUI mode: Paper size options

```
> tlmgr update -all
```

If this makes you nervous, first try

```
> tlmgr update -all -dry-run
```

or (less verbose):

```
> tlmgr update -list
```

This more complex example adds a collection, for the engine XeTEX, from a local directory:

```
> tlmgr -repository /local/mirror/tlnet install collection-xetex
```

It generates the following output (abridged):

```
install: collection-xetex
install: arabxetex
...
install: xetex
install: xetexconfig
install: xetex.i386-linux
running post install action for xetex
install: xetex-def
...
running mktexlsr
mktexlsr: Updating /usr/local/texlive/2015/texmf-dist/ls-R...
...
running fmtutil-sys --missing
...
Transcript written on xelatex.log.
fmtutil: /usr/local/texlive/2015/texmf-var/web2c/xetex/xelatex.fmt installed.
```

As you can see, tlmgr installs dependencies, and takes care of any necessary post-install actions, including updating the filename database and (re)generating formats. In the above, we generated new formats for XeTEX.

To describe a package (or collection or scheme):

```
> tlmgr show collection-latexextra
```

which produces output like this:

```
package:    collection-latexextra
category:   Collection
shortdesc:  LaTeX supplementary packages
longdesc:   A very large collection of add-on packages for LaTeX.
installed:  Yes
revision:   32768
```

Last and most important, for full documentation see http://tug.org/texlive/tlmgr.html, or:

```
> tlmgr -help
```

6 Notes on Windows

6.1 Windows-specific features

Under Windows, the installer does some extra things:

Menus and shortcuts. A new 'TEX Live' submenu of the Start menu is installed, which contains entries for some GUI programs (tlmgr, texdoctk, the PS_View (psv) PostScript previewer) and some documentation.

File associations. If enabled, TeXworks, Dviout and PS_view become either the default program for their respective filetypes, or get an entry in the 'Open with' right-click menus of those filetypes.

Bitmap to eps converter. Various bitmapped formats get an entry bitmap2eps in their 'Open with' right-click menu. Bitmap2eps is a simple script which lets sam2p or bmeps do the real work.

Automatic path adjustment. No manual configuration steps are required.

Uninstaller. The installer creates an entry under 'Add/Remove Programs' for TEX Live. The uninstall tab of the TEX Live Manager GUI refers to this.

6.2 Additional software included on Windows

To be complete, a TEX Live installation needs support packages that are not commonly found on a Windows machine. TEX Live provides the missing pieces. These programs are all installed as part of TEX Live only on Windows.

Perl and Ghostscript. Because of the importance of Perl and Ghostscript, TEX Live includes 'hidden' copies of these programs. TEX Live programs that need them know where to find them, but they don't betray their presence through environment variables or registry settings. They aren't full-scale installations, and shouldn't interfere with any system installations of Perl or Ghostscript.

PS_View. Also installed is PS_View, a PostScript and PDF viewer; see figure 12.

dviout. Also installed is dviout, a DVI viewer. At first, when you preview files with dviout, it will create fonts, because screen fonts were not installed. After a while, you will have created most of the fonts you use, and you will rarely see the font-creation window. More information can be found in the (highly recommended) on-line help.

TEXworks. TEXworks is a TEX-oriented editor with an integrated PDF viewer.

Command-line tools. A number of Windows ports of common Unix command-line programs are installed along with the usual TEX Live binaries. These include gzip, unzip and the utilities from the xpdf suite (pdfinfo, pdffonts, ...). The xpdf viewer itself is not available for Windows. Instead, you can download the Sumatra PDF viewer, which is based on xpdf, from http://blog.kowalczyk.info/software/sumatrapdf, among other options.

fc-list, fc-cache, ... The tools from the fontconfig library allow XeTEX to handle system fonts on Windows. You can use fc-list to determine the font names to pass to XeTEX's extended \font command. If necessary, run fc-cache first to update font information.

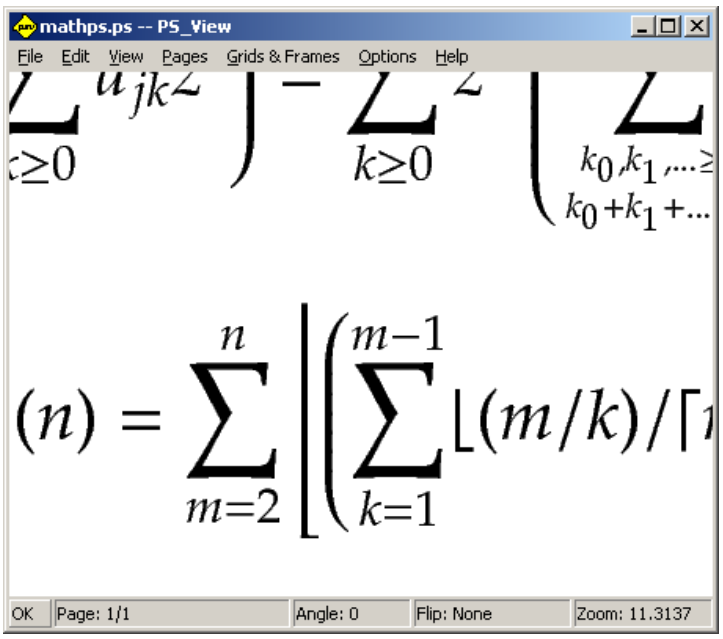


Figure 12: PS_View: very high magnifications available!

6.3 User Profile is Home

The Windows counterpart of a Unix home directory is the `%USERPROFILE%` directory. Under Windows XP, this is usually `C:\Documents and Settings\<username>`, and under Windows Vista and later it is `C:\Users\<username>`. In the `texmf.cnf` file, and Kpathsea in general, `~` will expand appropriately on both Windows and Unix.

6.4 The Windows registry

Windows stores nearly all configuration data in its registry. The registry contains a set of hierarchically organized keys, with several root keys. The most important ones for installation programs are `HKEY_CURRENT_USER` and `HKEY_LOCAL_MACHINE`, `HKCU` and `HKLM` in short. The `HKCU` part of the registry is in the user's home directory (see section 6.3). `HKLM` is normally in a subdirectory of the Windows directory.

In some cases, system information could be obtained from environment variables but for other information, for example the location of shortcuts, it is necessary to consult the registry. Setting environment variables permanently also requires registry access.

6.5 Windows permissions

In later versions of Windows, a distinction is made between regular users and administrators, where only the latter have free access to the entire operating system. We have made an effort to make TEX Live installable without administrative privileges.

If the installer is started with administrative permissions, there is an option to install for all users. If this option is chosen, shortcuts are created for all users, and the system environment is modified. Otherwise, shortcuts and menu entries are created for the current user, and the user environment is modified.

Regardless of administrator status, the default root of TEX Live proposed by the installer is always under `%SystemDrive%`. The installer always tests whether the root is writable for the current user.

A problem may arise if the user is not an administrator and TEX already exists in the search path. Since the effective path consists of the system path followed by the user path, the new TEX Live would never get precedence. As a safeguard, the installer creates a shortcut to the command-prompt in which the new TEX Live binary directory is prepended to the local search path. The new TEX Live will be always usable from within such a command-prompt. The shortcut for TEXworks, if installed, also prepends TEX Live to the search path, so it should also be immune to this path problem.

For Windows Vista and later, you should be aware that even if you are logged in as administrator, you need to explicitly ask for administrator privileges. In fact, there is not much point in logging in as

administrator. Instead, right-clicking on the program or shortcut that you want to run usually gives you a choice 'Run as administrator'.

6.5.1 Protecting the TEX Live directory

An admin install does *not* protect the TEX Live directory from tampering by regular users. This will have to be done in a separate step by setting a suitable ACL (Access Control List) for the directory. To this end, check out the `icacls` Windows command-line utility.

6.6 Increasing maximum memory on Windows and Cygwin

Windows and Cygwin (see section 3.1.4 for Cygwin installation specifics) users may find that they run out of memory when running some of the programs shipped with TEX Live. For example, asy might run out of memory if you try to allocate an array of 25,000,000 reals, and LuaTEX might run out of memory if you try to process a document with a lot of big fonts.

For Cygwin, you can increase the amount of available memory by following the instructions in the Cygwin User's Guide (http://www.cygwin.com/cygwin-ug-net/setup-maxmem.html).

For Windows, you have to create a file, say `moremem.reg`, with these four lines:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\Software\Cygwin]
"heap_chunk_in_mb"=dword:ffffff00
```

and then execute the command `regedit /s moremem.reg` as administrator. (If you want to change memory only for the current user instead of system-wide, use `HKEY_CURRENT_USER`.)

7 A user's guide to Web2C

Web2C is an integrated collection of TEX-related programs: TEX itself, METAFONT, MetaPost, BIBTEX, etc. It is the heart of TEX Live. The home page for Web2C, with the current manual and more, is http://tug.org/web2c.

A bit of history: The original implementation was by Tomas Rokicki who, in 1987, developed a first TEX-to-C system based on change files under Unix, which were primarily the original work of Howard Trickey and Pavel Curtis. Tim Morgan became the maintainer of the system, and during this period the name changed to Web-to-C. In 1990, Karl Berry took over the work, assisted by dozens of additional contributors, and in 1997 he handed the baton to Olaf Weber, who returned it to Karl in 2006.

The Web2C system runs on Unix, 32-bit Windows systems, Mac OS X, and other operating systems. It uses Knuth's original sources for TEX and other basic programs written in the `WEB` literate programming system and translates them into C source code. The core TEX programs handled in this way are:

bibtex Maintaining bibliographies.

dvicopy Expands virtual font references in DVI files.

dvitomp DVI to MPX (MetaPost pictures).

dvitype DVI to human-readable text.

gftodvi Generic font proofsheets.

gftopk Generic to packed fonts.

gftype GF to human-readable text.

mf Creating typeface families.

mft Prettyprinting METAFONT source.

mpost Creating technical diagrams.

patgen Creating hyphenation patterns.

pktogf Packed to generic fonts.

pktype PK to human-readable text.

pltotf Plain text property list to TFM.

pooltype Display WEB pool files.

tangle WEB to Pascal.

tex Typesetting.

tftopl TFM to plain text property list.

vftovp Virtual font to virtual property list.

vptovf Virtual property list to virtual font.

weave WEB to TEX.

The precise functions and syntax of these programs are described in the documentation of the individual packages and of Web2C itself. However, knowing a few principles governing the whole family of programs will help you take advantage of your Web2C installation.

All programs honor these standard GNU options:

`--help` print basic usage summary.

`--verbose` print detailed progress report.

`--version` print version information, then exit.

For locating files the Web2C programs use the path searching library Kpathsea (http://tug.org/kpathsea). This library uses a combination of environment variables and a configuration files to optimize searching the (huge) collection of TEX files. Web2C can look at many directory trees simultaneously, which is useful in maintaining TEX's standard distribution and local and personal extensions in distinct trees. To speed up file searches, the root of each tree has a file `ls-R`, containing an entry showing the name and relative pathname for all files under that root.

7.1 Kpathsea path searching

Let us first describe the generic path searching mechanism of the Kpathsea library.

We call a *search path* a colon- or semicolon-separated list of *path elements*, which are basically directory names. A search path can come from (a combination of) many sources. To look up a file '`my-file`' along a path '`.:/dir`', Kpathsea checks each element of the path in turn: first `./my-file`, then `/dir/my-file`, returning the first match (or possibly all matches).

In order to adapt optimally to all operating systems' conventions, on non-Unix systems Kpathsea can use filename separators different from colon ('`:`') and slash ('`/`').

To check a particular path element p, Kpathsea first checks if a prebuilt database (see "Filename database" on page 28) applies to p, i.e., if the database is in a directory that is a prefix of p. If so, the path specification is matched against the contents of the database.

If the database does not exist, or does not apply to this path element, or contains no matches, the filesystem is searched (if this was not forbidden by a specification starting with '`!!`' and if the file being searched for must exist). Kpathsea constructs the list of directories that correspond to this path element, and then checks in each for the file being sought.

The "file must exist" condition comes into play with '`.vf`' files and input files read by TEX's `\openin` command. Such files may not exist (e.g., `cmr10.vf`), and so it would be wrong to search the disk for them. Therefore, if you fail to update `ls-R` when you install a new '`.vf`' file, it will never be found. Each path element is checked in turn: first the database, then the disk. If a match is found, the search stops and the result is returned.

Although the simplest and most common path element is a directory name, Kpathsea supports additional features in search paths: layered default values, environment variable names, config file values, users' home directories, and recursive subdirectory searching. Thus, we say that Kpathsea *expands* a path element, meaning it transforms all the specifications into basic directory name or names. This is described in the following sections in the same order as it takes place.

Note that if the filename being searched for is absolute or explicitly relative, i.e., starts with '`/`' or '`./`' or '`../`', Kpathsea simply checks if that file exists.

7.1.1 Path sources

A search path can come from many sources. In the order in which Kpathsea uses them:

1. A user-set environment variable, for instance, `TEXINPUTS`. Environment variables with a period and a program name appended override; e.g., if '`latex`' is the name of the program being run, then `TEXINPUTS.latex` will override `TEXINPUTS`.

2. A program-specific configuration file, for example, a line '`S /a:/b`' in `dvips`'s `config.ps`.
3. A Kpathsea configuration file `texmf.cnf`, containing a line like '`TEXINPUTS=/c:/d`' (see below).
4. The compile-time default.

You can see each of these values for a given search path by using the debugging options (see "Debugging actions" on page 30).

7.1.2 Config files

Kpathsea reads *runtime configuration files* named `texmf.cnf` for search path and other definitions. The search path used to look for these files is named `TEXMFCNF`, but we do not recommend setting this (or any) environment variable.

Instead, normal installation results in a file `.../2015/texmf.cnf`. If you must make changes to the defaults (not normally necessary), this is the place to put them. The main configuration file is in `.../2015/texmf-dist/web2c/texmf.cnf`. You should not edit this latter file, as your changes will be lost when the distributed version is updated.

All `texmf.cnf` files in the search path will be read and definitions in earlier files override those in later files. For example, with a search path of `.:$TEXMF`, values from `./texmf.cnf` override those from `$TEXMF/texmf.cnf`.

- Comments start with `%` and continue to the end of the line.
- Blank lines are ignored.
- A `\` at the end of a line acts as a continuation character, i.e., the next line is appended. Whitespace at the beginning of continuation lines is not ignored.
- Each remaining line has the form:

  variable`[.`*progname*`]` `[=]` *value*

  where the '`=`' and surrounding whitespace are optional.
- The *variable* name may contain any character other than whitespace, '`=`', or '`.`', but sticking to '`A-Za-z_`' is safest.
- If '`.`*progname*' is present, the definition only applies if the program that is running is named *progname* or *progname*`.exe`. This allows different flavors of TEX to have different search paths, for example.
- *value* may contain any characters except `%` and '`@`'. The `$`*var*`.`*prog* feature is not available on the right-hand side; instead, you must use an additional variable. A '`;`' in *value* is translated to '`:`' if running under Unix; this is useful to be able to have a single `texmf.cnf` for Unix, MS-DOS and Windows systems.
- All definitions are read before anything is expanded, so variables can be referenced before they are defined.

A configuration file fragment illustrating most of these points is shown below:

```
TEXMF              = {$TEXMFLOCAL,!!$TEXMFMAIN}
TEXINPUTS.latex    = .;$TEXMF/tex/{latex,generic;}//
TEXINPUTS.fontinst = .;$TEXMF/tex//;$TEXMF/fonts/afm//
% e-TeX related files
TEXINPUTS.elatex   = .;$TEXMF/{etex,tex}/{latex,generic;}//
TEXINPUTS.etex     = .;$TEXMF/{etex,tex}/{eplain,plain,generic;}//
```

7.1.3 Path expansion

Kpathsea recognizes certain special characters and constructions in search paths, similar to those available in Unix shells. As a general example, the complex path, `~$USER/{foo,bar}//baz`, expands to all subdirectories under directories `foo` and `bar` in `$USER`'s home directory that contain a directory or file `baz`. These expansions are explained in the sections below.

7.1.4 Default expansion

If the highest-priority search path (see "Path sources" on page 25) contains an *extra colon* (i.e., leading, trailing, or doubled), Kpathsea inserts at that point the next-highest-priority search path that is defined. If that inserted path has an extra colon, the same happens with the next highest. For example, given an environment variable setting

```
> setenv TEXINPUTS /home/karl:
```

and a `TEXINPUTS` value from `texmf.cnf` of

```
.:$TEXMF//tex
```

then the final value used for searching will be:

```
/home/karl:.:$TEXMF//tex
```

Since it would be useless to insert the default value in more than one place, Kpathsea changes only one extra '`:`' and leaves any others in place. It checks first for a leading '`:`', then a trailing '`:`', then a doubled '`:`'.

7.1.5 Brace expansion

A useful feature is brace expansion, which means that, for instance, `v{a,b}w` expands to `vaw:vbw`. Nesting is allowed. This is used to implement multiple TEX hierarchies, by assigning a brace list to `$TEXMF`. For example, in `texmf.cnf`, a definition like this (simplified for this example) is made:

```
TEXMF = {$TEXMFVAR,$TEXMFHOME,!!$TEXMFLOCAL,!!$TEXMFDIST}
```

We can then use this to define, for example, the TEX input path:

```
TEXINPUTS = .;$TEXMF/tex//
```

which means that, after looking in the current directory, the `$TEXMFVAR/tex`, `$TEXMFHOME/tex`, `$TEXMFLOCAL/tex` and `$TEXMFDIST/tex` trees *only*) will be searched (the last two using `ls-R` data base files). It is a convenient way for running two parallel TEX structures, one "frozen" (on a CD, for instance) and the other being continuously updated with new versions as they become available. By using the `$TEXMF` variable in all definitions, one is sure to always search the up-to-date tree first.

7.1.6 Subdirectory expansion

Two or more consecutive slashes in a path element following a directory *d* is replaced by all subdirectories of *d*: first those subdirectories directly under *d*, then the subsubdirectories under those, and so on. At each level, the order in which the directories are searched is *unspecified*.

If you specify any filename components after the '`//`', only subdirectories with matching components are included. For example, '`/a//b`' expands into directories `/a/1/b`, `/a/2/b`, `/a/1/1/b`, and so on, but not `/a/b/c` or `/a/1`.

Multiple '`//`' constructs in a path are possible, but '`//`' at the beginning of a path is ignored.

7.1.7 List of special characters and their meaning: a summary

The following list summarizes the special characters in Kpathsea configuration files.

`:` Separator in path specification; at the beginning or the end of a path it substitutes the default path expansion.

`;` Separator on non-Unix systems (acts like `:`).

`$` Variable expansion.

`~` Represents the user's home directory.

`{...}` Brace expansion.

`//` Subdirectory expansion (can occur anywhere in a path, except at its beginning).

`%` Start of comment.

`\` Continuation character (allows multi-line entries).

`!!` Search *only* database to locate file, *do not* search the disk.

7.2 Filename databases

Kpathsea goes to some lengths to minimize disk accesses for searches. Nevertheless, at installations with enough directories, searching each possible directory for a given file can take an excessively long time (this is especially true if many hundreds of font directories have to be traversed.) Therefore, Kpathsea can use an externally-built plain text "database" file named `ls-R` that maps files to directories, thus avoiding the need to exhaustively search the disk.

A second database file `aliases` allows you to give additional names to the files listed in `ls-R`. This can be helpful to confirm to DOS 8.3 filename conventions in source files.

7.2.1 The filename database

As explained above, the name of the main filename database must be `ls-R`. You can put one at the root of each TEX hierarchy in your installation that you wish to be searched (`$TEXMF` by default). Kpathsea looks for `ls-R` files along the `TEXMFDBS` path.

The recommended way to create and maintain '`ls-R`' is to run the `mktexlsr` script included with the distribution. It is invoked by the various '`mktex`'... scripts. In principle, this script just runs the command

`cd` */your/texmf/root* `&& \ls -1LAR ./ >ls-R`

presuming your system's `ls` produces the right output format (GNU `ls` is all right). To ensure that the database is always up-to-date, it is easiest to rebuild it regularly via `cron`, so that it is automatically updated when the installed files change, such as after installing or updating a LATEX package.

If a file is not found in the database, by default Kpathsea goes ahead and searches the disk. If a particular path element begins with '`!!`', however, *only* the database will be searched for that element, never the disk.

7.2.2 kpsewhich: Standalone path searching

The `kpsewhich` program exercises path searching independent of any particular application. This can be useful as a sort of `find` program to locate files in TEX hierarchies (this is used heavily in the distributed '`mktex`'... scripts).

`> kpsewhich` *option*`...` *filename*`...`

The options specified in *option* start with either '`-`' or '`--`', and any unambiguous abbreviation is accepted.

Kpathsea looks up each non-option argument on the command line as a filename, and returns the first file found. There is no option to return all the files with a particular name (you can run the Unix '`find`' utility for that).

The most common options are described next.

`--dpi=`*num* Set the resolution to *num*; this only affects '`gf`' and '`pk`' lookups. '`-D`' is a synonym, for compatibility with `dvips`. Default is 600.

`--format=`*name*
Set the format for lookup to *name*. By default, the format is guessed from the filename. For formats which do not have an associated unambiguous suffix, such as MetaPost support files and `dvips` configuration files, you have to specify the name as known to Kpathsea, such as `tex` or `enc files`. Run `kpsewhich --help` for a list.

`--mode=`*string*
Set the mode name to *string*; this only affects '`gf`' and '`pk`' lookups. No default: any mode will be found.

`--must-exist`
Do everything possible to find the files, notably including searching the disk. By default, only the `ls-R` database is checked, in the interest of efficiency.

`--path=`*string*
Search along the path *string* (colon-separated as usual), instead of guessing the search path from the filename. '`//`' and all the usual expansions are supported. The options '`--path`' and '`--format`' are mutually exclusive.

`--progname=`*name*
: Set the program name to *name*. This can affect the search paths via the *.progname* feature. The default is kpsewhich.

`--show-path=`*name*
: shows the path used for file lookups of file type *name*. Either a filename extension (`.pk`, `.vf`, etc.) or a name can be used, just as with '`--format`' option.

`--debug=`*num*
: sets the debugging options to *num*.

7.2.3 Examples of use

Let us now have a look at Kpathsea in action. Here's a straightforward search:

```
> kpsewhich article.cls
   /usr/local/texmf-dist/tex/latex/base/article.cls
```

We are looking for the file `article.cls`. Since the '`.cls`' suffix is unambiguous we do not need to specify that we want to look for a file of type `tex` (TEX source file directories). We find it in the subdirectory `tex/latex/base` below the '`texmf-dist`' TEX Live directory. Similarly, all of the following are found without problems thanks to their unambiguous suffix.

```
> kpsewhich array.sty
   /usr/local/texmf-dist/tex/latex/tools/array.sty
> kpsewhich latin1.def
   /usr/local/texmf-dist/tex/latex/base/latin1.def
> kpsewhich size10.clo
   /usr/local/texmf-dist/tex/latex/base/size10.clo
> kpsewhich small2e.tex
   /usr/local/texmf-dist/tex/latex/base/small2e.tex
> kpsewhich tugboat.bib
   /usr/local/texmf-dist/bibtex/bib/beebe/tugboat.bib
```

By the way, that last is a BIBTEX bibliography database for *TUGboat* articles.

```
> kpsewhich cmr10.pk
```

Font bitmap glyph files of type `.pk` are used by display programs like dvips and xdvi. Nothing is returned in this case since there are no pre-generated Computer Modern '`.pk`' files in TEX Live — the Type 1 variants are used by default.

```
> kpsewhich wsuipa10.pk
   /usr/local/texmf-var/fonts/pk/ljfour/public/wsuipa/wsuipa10.600pk
```

For these fonts (a phonetic alphabet from the University of Washington) we had to generate '`.pk`' files, and since the default METAFONT mode on our installation is `ljfour` with a base resolution of 600 dpi (dots per inch), this instantiation is returned.

```
> kpsewhich -dpi=300 wsuipa10.pk
```

In this case, when specifying that we are interested in a resolution of 300 dpi (`-dpi=300`) we see that no such font is available on the system. A program like dvips or xdvi would go off and actually build the required `.pk` files using the script mktexpk.

Next we turn our attention to dvips's header and configuration files. We first look at one of the commonly used files, the general prologue `tex.pro` for TEX support, before turning our attention to the generic configuration file (`config.ps`) and the PostScript font map `psfonts.map` — as of 2004, map and encoding files have their own search paths and new location in `texmf` trees. As the '`.ps`' suffix is ambiguous we have to specify explicitly which type we are considering (`dvips config`) for the file `config.ps`.

```
> kpsewhich tex.pro
   /usr/local/texmf/dvips/base/tex.pro
> kpsewhich --format="dvips config" config.ps
   /usr/local/texmf/dvips/config/config.ps
> kpsewhich psfonts.map
   /usr/local/texmf/fonts/map/dvips/updmap/psfonts.map
```

We now take a closer look at the URW Times PostScript support files. The prefix for these in the standard font naming scheme is '`utm`'. The first file we look at is the configuration file, which contains the name of the map file:

```
> kpsewhich --format="dvips config" config.utm
   /usr/local/texmf-dist/dvips/psnfss/config.utm
```

The contents of that file is

```
p +utm.map
```

which points to the file `utm.map`, which we want to locate next.

```
> kpsewhich utm.map
   /usr/local/texmf-dist/fonts/map/dvips/times/utm.map
```

This map file defines the file names of the Type 1 PostScript fonts in the URW collection. Its contents look like (we only show part of the lines):

```
utmb8r  NimbusRomNo9L-Medi    ... <utmb8a.pfb
utmbi8r NimbusRomNo9L-MediItal... <utmbi8a.pfb
utmr8r  NimbusRomNo9L-Regu    ... <utmr8a.pfb
utmri8r NimbusRomNo9L-ReguItal... <utmri8a.pfb
utmbo8r NimbusRomNo9L-Medi    ... <utmb8a.pfb
utmro8r NimbusRomNo9L-Regu    ... <utmr8a.pfb
```

Let us, for instance, take the Times Roman instance `utmr8a.pfb` and find its position in the `texmf` directory tree with a search for Type 1 font files:

```
> kpsewhich utmr8a.pfb
   /usr/local/texmf-dist/fonts/type1/urw/times/utmr8a.pfb
```

It should be evident from these examples how you can easily locate the whereabouts of a given file. This is especially important if you suspect that the wrong version of a file is picked up somehow, since kpsewhich will show you the first file encountered.

7.2.4 Debugging actions

Sometimes it is necessary to investigate how a program resolves file references. To make this practical, Kpathsea offers various levels of debugging output:

- `1` `stat` calls (disk lookups). When running with an up-to-date `ls-R` database this should almost give no output.
- `2` References to hash tables (such as `ls-R` databases, map files, configuration files).
- `4` File open and close operations.
- `8` General path information for file types searched by Kpathsea. This is useful to find out where a particular path for the file was defined.
- `16` Directory list for each path element (only relevant for searches on disk).
- `32` File searches.
- `64` Variable values.

A value of `-1` will set all the above options; in practice, this is usually the most convenient.

Similarly, with the dvips program, by setting a combination of debug switches, one can follow in detail where files are being picked up from. Alternatively, when a file is not found, the debug trace shows in which directories the program looks for the given file, so that one can get an indication what the problem is.

Generally speaking, as most programs call the Kpathsea library internally, one can select a debug option by using the `KPATHSEA_DEBUG` environment variable, and setting it to (a combination of) values as described in the above list.

(Note for Windows users: it is not easy to redirect all messages to a file in this system. For diagnostic purposes you can temporarily `SET KPATHSEA_DEBUG_OUTPUT=err.log`).

Let us consider, as an example, a small LaTeX source file, `hello-world.tex`, which contains the following input.

```
debug:start search(file=texmf.cnf, must_exist=1, find_all=1,
  path=.:/usr/local/bin/texlive:/usr/local/bin:
       /usr/local/bin/texmf/web2c:/usr/local:
       /usr/local/texmf/web2c:/.:/./teTeX/TeX/texmf/web2c:).
kdebug:start search(file=ls-R, must_exist=1, find_all=1,
  path=~/tex:/usr/local/texmf).
kdebug:search(ls-R) =>/usr/local/texmf/ls-R
kdebug:start search(file=aliases, must_exist=1, find_all=1,
  path=~/tex:/usr/local/texmf).
kdebug:search(aliases) => /usr/local/texmf/aliases
kdebug:start search(file=config.ps, must_exist=0, find_all=0,
  path=.:~/tex:!!/usr/local/texmf/dvips//).
kdebug:search(config.ps) => /usr/local/texmf/dvips/config/config.ps
kdebug:start search(file=/root/.dvipsrc, must_exist=0, find_all=0,
  path=.:~/tex:!!/usr/local/texmf/dvips//).
search(file=/home/goossens/.dvipsrc, must_exist=1, find_all=0,
  path=.:~/tex/dvips//:!!/usr/local/texmf/dvips//).
kdebug:search($HOME/.dvipsrc) =>
kdebug:start search(file=config.cms, must_exist=0, find_all=0,
  path=.:~/tex/dvips//:!!/usr/local/texmf/dvips//).
kdebug:search(config.cms)
=>/usr/local/texmf/dvips/cms/config.cms
```

Figure 13: Finding configuration files

```
kdebug:start search(file=texc.pro, must\_exist=0, find\_all=0,
  path=.:~/tex/dvips//:!!/usr/local/texmf/dvips//:
       ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(texc.pro) => /usr/local/texmf/dvips/base/texc.pro
```

Figure 14: Finding the prolog file

```
kdebug:start search(file=cmr10.tfm, must\_exist=1, find\_all=0,
  path=.:~/tex/fonts/tfm//:!!/usr/local/texmf/fonts/tfm//:
       /var/tex/fonts/tfm//).
kdebug:search(cmr10.tfm) => /usr/local/texmf/fonts/tfm/public/cm/cmr10.tfm
kdebug:start search(file=texps.pro, must\_exist=0, find\_all=0,
   ...
<texps.pro>
kdebug:start search(file=cmr10.pfb, must\_exist=0, find\_all=0,
  path=.:~/tex/dvips//:!!/usr/local/texmf/dvips//:
       ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(cmr10.pfb) => /usr/local/texmf/fonts/type1/public/cm/cmr10.pfb
<cmr10.pfb>[1]
```

Figure 15: Finding the font file

```
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

This little file only uses the font `cmr10`, so let us look at how dvips prepares the PostScript file (we want to use the Type 1 version of the Computer Modern fonts, hence the option `-Pcms`).

```
> dvips -d4100 hello-world -Pcms -o
```

In this case we have combined dvips's debug class 4 (font paths) with Kpathsea's path element expansion (see the dvips reference manual). The output (slightly rearranged) appears in Figure 13.

dvips starts by locating its working files. First, `texmf.cnf` is found, which gives the definitions of the search paths for the other files, then the file database `ls-R` (to optimize file searching) and the file `aliases`, which makes it possible to declare several names (e.g., a short DOS-like 8.3 and a more natural longer version) for the same file. Then dvips goes on to find the generic configuration file `config.ps` before looking for the customization file `.dvipsrc` (which, in this case is *not found*). Finally, dvips locates the config file for the Computer Modern PostScript fonts `config.cms` (this was initiated with

the -Pcms option on the dvips command). This file contains the list of the map files which define the relation between the TEX, PostScript and file system names of the fonts.

```
> more /usr/local/texmf/dvips/cms/config.cms
   p +ams.map
   p +cms.map
   p +cmbkm.map
   p +amsbkm.map
```

dvips thus goes on to find all these files, plus the generic map file psfonts.map, which is always loaded (it contains declarations for commonly used PostScript fonts; see the last part of Section 7.2.3 for more details about PostScript map file handling).

At this point dvips identifies itself to the user:

```
This is dvips(k) 5.92b Copyright 2002 Radical Eye Software (www.radicaleye.com)
```

Then it goes on to look for the prolog file texc.pro:

```
kdebug:start search(file=texc.pro, must_exist=0, find_all=0,
  path=.:~/tex/dvips//:!!/usr/local/texmf/dvips//:
       ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(texc.pro) => /usr/local/texmf/dvips/base/texc.pro
```

After having found the file in question, dvips outputs the date and time, and informs us that it will generate the file hello-world.ps, then that it needs the font file cmr10, and that the latter is declared as "resident" (no bitmaps needed):

```
TeX output 1998.02.26:1204' -> hello-world.ps
Defining font () cmr10 at 10.0pt
Font cmr10 <CMR10> is resident.
```

Now the search is on for the file cmr10.tfm, which is found, then a few more prolog files (not shown) are referenced, and finally the Type 1 instance cmr10.pfb of the font is located and included in the output file (see last line).

```
kdebug:start search(file=cmr10.tfm, must_exist=1, find_all=0,
  path=.:~/tex/fonts/tfm//:!!/usr/local/texmf/fonts/tfm//:
       /var/tex/fonts/tfm//).
kdebug:search(cmr10.tfm) => /usr/local/texmf/fonts/tfm/public/cm/cmr10.tfm
kdebug:start search(file=texps.pro, must_exist=0, find_all=0,
   ...
<texps.pro>
kdebug:start search(file=cmr10.pfb, must_exist=0, find_all=0,
  path=.:~/tex/dvips//:!!/usr/local/texmf/dvips//:
       ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(cmr10.pfb) => /usr/local/texmf/fonts/type1/public/cm/cmr10.pfb
<cmr10.pfb>[1]
```

7.3 Runtime options

Another useful feature of Web2C is its possibility to control a number of memory parameters (in particular, array sizes) via the runtime file texmf.cnf read by Kpathsea. The memory settings can be found in Part 3 of that file in the TEX Live distribution. The more important are:

main_memory Total words of memory available, for TEX, METAFONT and MetaPost. You must make a new format file for each different setting. For instance, you could generate a "huge" version of TEX, and call the format file hugetex.fmt. Using the standard way of specifying the program name used by Kpathsea, the particular value of the main_memory variable will then be read from texmf.cnf.

extra_mem_bot Extra space for "large" TEX data structures: boxes, glue, breakpoints, etc. Especially useful if you use PICTEX.

font_mem_size Number of words for font information available for TEX. This is more or less the total size of all TFM files read.

`hash_extra` Additional space for the hash table of control sequence names. Only ≈10,000 control sequences can be stored in the main hash table; if you have a large book with numerous cross-references, this might not be enough. The default value of `hash_extra` is `50000`.

Of course, this facility is no substitute for truly dynamic arrays and memory allocation, but since these are extremely difficult to implement in the present TEX source, these runtime parameters provide a practical compromise allowing some flexibility.

8 Acknowledgements

TEX Live is a joint effort by virtually all of the TEX user groups. This edition of TEX Live was overseen by Karl Berry. The other principal contributors, past and present, are listed below.

- The English, German, Dutch, and Polish TEX user groups (TUG, DANTE e.V., NTG, and GUST, respectively), which provide the necessary technical and administrative infrastructure. Please join the TEX user group near you! (See http://tug.org/usergroups.html.)
- The CTAN team, notably Robin Fairbairns and Rainer Schöpf, which distributes the TEX Live images and provides the common infrastructure for package updates, upon which TEX Live depends.
- Nelson Beebe, for making many platforms available to TEX Live developers, and his own comprehensive testing.
- John Bowman, for making many changes to his advanced graphics program Asymptote to make it work in TEX Live.
- Peter Breitenlohner and the ε-TEX team for the stable foundation of future TEX's, and Peter specifically for stellar help with GNU autotools usage throughout TEX Live.
- Jin-Hwan Cho and all of the DVIPDFM*x* team, for their excellent driver and responsiveness to configuration issues.
- Thomas Esser, without whose marvelous teTEX package TEX Live would have never existed.
- Michel Goossens, who co-authored the original documentation.
- Eitan Gurari, whose TEX4ht is used to create the HTML version of this documentation, and who worked tirelessly to improve it at short notice, every year. Eitan prematurely passed away in June 2009, and we dedicate this documentation to his memory.
- Hans Hagen, for much testing and making his ConTEXt format (http://pragma-ade.com) work within TEX Live's framework.
- Hàn Thế Thành, Martin Schröder, and the pdfTEX team (http://pdftex.org) for continuing enhancements of TEX's abilities.
- Hartmut Henkel, for significant development contributions to pdfTEX LuaTEX, and more.
- Taco Hoekwater, for major renewed development efforts on MetaPost and (Lua)TEX (http://luatex.org) itself, incorporating ConTEXt into TEX Live, giving Kpathsea multi-threaded functionality, and much more.
- Paweł Jackowski, for the Windows installer tlpm, and Tomasz Łuczak, for tlpmgui, used in past releases.
- Akira Kakuto, for providing the Windows binaries from his W32TEX distribution for Japanese TEX (http://w32tex.org), and many other development contributions.
- Jonathan Kew, for developing the remarkable XeTEX engine and taking the time and trouble to integrate it in TEX Live, as well as the initial version of the MacTEX installer, and also for our recommended front-end TEXworks.
- Dick Koch, for maintaining MacTEX (http://tug.org/mactex) in very close tandem with TEX Live, and for his great good cheer in doing so.
- Reinhard Kotucha, for major contributions to the TEX Live 2008 infrastructure and installer, as well as Windows research efforts, the `getnonfreefonts` script, and more.
- Siep Kroonenberg, also for major contributions to the TEX Live 2008 infrastructure and installer, especially on Windows, and for the bulk of work updating this manual describing those features.
- Mojca Miklavec, for much help with ConTEXt.
- Heiko Oberdiek, for the epstopdf package and many others, compressing the huge pst-geo data files so we could include them, and most of all, for his remarkable work on hyperref.
- Petr Olšak, who coordinated and checked all the Czech and Slovak material very carefully.
- Toshio Oshima, for his dviout previewer for Windows.
- Manuel Pégourié-Gonnard, for helping with package updates, documentation improvements, and texdoc development.

- Fabrice Popineau, for the original Windows support in TEX Live and work on the French documentation.
- Norbert Preining, the principal architect of the TEX Live 2008 infrastructure and installer, and also for coordinating the Debian version of TEX Live (together with Frank Küster), making many suggestions along the way.
- Sebastian Rahtz, for originally creating TEX Live and maintaining it for many years.
- Phil Taylor, for setting up the BitTorrent downloads.
- Tomasz Trzeciak, for wide-ranging help with Windows.
- Vladimir Volovich, for substantial help with porting and other maintenance issues, and especially for making it feasible to include xindy.
- Staszek Wawrykiewicz, the principal tester for all of TEX Live, and coordinator of the many major Polish contributions: fonts, Windows installation, and more.
- Olaf Weber, for his patient maintenance of Web2C.
- Gerben Wierda, for creating and maintaining the original Mac OS X support.
- Graham Williams, the originator of the TEX Catalogue.

Builders of the binaries: Ettore Aldrovandi (i386-solaris, x86_64-solaris), Marc Baudoin (amd64-netbsd, i386-netbsd), Karl Berry (i386-linux), Peter Breitenlohner (x86_64-linux), Ken Brown (i386-cygwin, x86_64-cygwin), Simon Dales (armhf-linux), Akira Kakuto (win32), Dick Koch (universal-darwin, x86_64-darwin), Nikola Lečić (amd64-freebsd, i386-freebsd), Mojca Miklavec (mipsel-linux), sparc-solaris), Norbert Preining (alpha-linux), Thomas Schmitz (powerpc-linux), Boris Veytsman (armel-linux). For information on the TEX Live build process, see http://tug.org/texlive/build.html.

Translators of this manual: Boris Veytsman (Russian), Jjgod Jiang, Jinsong Zhao, Yue Wang, & Helin Gai (Chinese), Uwe Ziegenhagen (German), Manuel Pégourié-Gonnard (French), Marco Pallante (Italian), Nikola Lečić (Serbian), Petr Sojka & Jan Busa (Czech/Slovak), Staszek Wawrykiewicz (Polish). The TEX Live documentation web page is http://tug.org/texlive/doc.html.

Of course the most important acknowledgement must go to Donald Knuth, first for inventing TEX, and then for giving it to the world.

9 Release history

9.1 Past

Discussion began in late 1993 when the Dutch TEX Users Group was starting work on its 4AllTEX CD for MS-DOS users, and it was hoped at that time to issue a single, rational, CD for all systems. This was too ambitious a target for the time, but it did spawn not only the very successful 4AllTEX CD, but also the TUG Technical Council working group on a *TEX Directory Structure* (http://tug.org/tds), which specified how to create consistent and manageable collections of TEX support files. A complete draft of the TDS was published in the December 1995 issue of *TUGboat*, and it was clear from an early stage that one desirable product would be a model structure on CD. The distribution you now have is a very direct result of the working group's deliberations. It was also clear that the success of the 4AllTEX CD showed that Unix users would benefit from a similarly easy system, and this is the other main strand of TEX Live.

We first undertook to make a new Unix-based TDS CD in the autumn of 1995, and quickly identified Thomas Esser's teTEX as the ideal setup, as it already had multi-platform support and was built with portability across file systems in mind. Thomas agreed to help, and work began seriously at the start of 1996. The first edition was released in May 1996. At the start of 1997, Karl Berry completed a major new release of Web2c, which included nearly all the features which Thomas Esser had added in teTEX, and we decided to base the 2nd edition of the CD on the standard Web2C, with the addition of teTEX's `texconfig` script. The 3rd edition of the CD was based on a major revision of Web2C, 7.2, by Olaf Weber; at the same time, a new revision of teTEX was being made, and TEX Live included almost all of its features. The 4th edition followed the same pattern, using a new version of teTEX, and a new release of Web2C (7.3). The system now included a complete Windows setup.

For the 5th edition (March 2000) many parts of the CD were revised and checked, updating hundreds of packages. Package details were stored in XML files. But the major change for TEX Live 5 was that all non-free software was removed. Everything in TEX Live is now intended to be compatible with the Debian Free Software Guidelines (http://www.debian.org/intro/free); we have done our best to check the license conditions of all packages, but we would very much appreciate hearing of any mistakes.

The 6th edition (July 2001) had much more material updated. The major change was a new install concept: the user could select a more exact set of needed collections. Language-related collections were completely reorganized, so selecting any of them installs not only macros, fonts, etc., but also prepares an appropriate `language.dat`.

The 7th edition of 2002 had the notable addition of Mac OS X support, and the usual myriad of updates to all sorts of packages and programs. An important goal was integration of the source back with teTEX, to correct the drift apart in versions 5 and 6.

9.1.1 2003

In 2003, with the continuing flood of updates and additions, we found that TEX Live had grown so large it could no longer be contained on a single CD, so we split it into three different distributions (see section 2.1, p. 4). In addition:

- At the request of the LATEX team, we changed the standard latex and pdflatex commands to now use ε-TEX (see p. 6).
- The new Latin Modern fonts were included (and are recommended).
- Support for Alpha OSF was removed (HPUX support was removed previously), since no one had (or volunteered) hardware available on which to compile new binaries.
- Windows setup was substantially changed; for the first time an integrated environment based on XEmacs was introduced.
- Important supplementary programs for Windows (Perl, Ghostscript, ImageMagick, Ispell) are now installed in the TEX Live installation directory.
- Font map files used by dvips, dvipdfm and pdftex are now generated by the new program updmap and installed into `texmf/fonts/map`.
- TEX, METAFONT, and MetaPost now, by default, output most input characters (32 and above) as themselves in output (e.g., `\write`) files, log files, and the terminal, i.e., *not* translated using the ^^ notation. In TEX Live 7, this translation was dependent on the system locale settings; now, locale settings do not influence the TEX programs' behavior. If for some reason you need the ^^ output, rename the file `texmf/web2c/cp8bit.tcx`. (Future releases will have cleaner ways to control this.)
- This documentation was substantially revised.
- Finally, since the edition numbers had grown unwieldy, the version is now simply identified by the year: TEX Live 2003.

9.1.2 2004

2004 saw many changes:

- If you have locally-installed fonts which use their own `.map` or (much less likely) `.enc` support files, you may need to move those support files.

  `.map` files are now searched for in subdirectories of `fonts/map` only (in each `texmf` tree), along the `TEXFONTMAPS` path. Similarly, `.enc` files are now searched for in subdirectories of `fonts/enc` only, along the `ENCFONTS` path. updmap will attempt to warn about problematic files.

  For methods of handling this and other information, please see http://tug.org/texlive/mapenc.html.

- The TEX Collection has been expanded with the addition of a MiKTEX-based installable CD, for those who prefer that implementation to Web2C. See section 2 (p. 4).

- Within TEX Live, the single large `texmf` tree of previous releases has been replaced by three: `texmf`, `texmf-dist`, and `texmf-doc`. See section 2.2 (p. 4), and the `README` files for each.

- All TEX-related input files are now collected in the `tex` subdirectory of `texmf*` trees, rather than having separate sibling directories `tex`, `etex`, `pdftex`, `pdfetex`, etc. See `texmf-dist/doc/generic/tds/tds.html#Extensions`.

- Helper scripts (not meant to be invoked by users) are now located in a new `scripts` subdirectory of `texmf*` trees, and can be searched for via `kpsewhich -format=texmfscripts`. So if you have programs which call such scripts, they'll need to be adjusted. See `texmf-dist/doc/generic/tds/tds.html#Scripts`.

- Almost all formats leave most characters printable as themselves via the "translation file" `cp227.tcx`, instead of translating them with the `^^` notation. Specifically, characters at positions 32–256, plus tab, vertical tab, and form feed are considered printable and not translated. The exceptions are plain TEX (only 32–126 printable), ConTEXt (0–255 printable), and the Ω-related formats. This default behavior is almost the same as in TEX Live 2003, but it's implemented more cleanly, with more possibilities for customization. See `texmf-dist/doc/web2c/web2c.html#TCX-files`. (By the way, with Unicode input, TEX may output partial character sequences when showing error contexts, since it is byte-oriented.)

- pdfetex is now the default engine for all formats except (plain) tex itself. (Of course it generates DVI when run as latex, etc.) This means, among other things, that the microtypographic features of pdftex are available in LATEX, ConTEXt, etc., as well as the ε-TEX features (`texmf-dist/doc/etex/base/`).

  It also means it's *more important than ever* to use the ifpdf package (works with both plain and LATEX) or equivalent code, because simply testing whether `\pdfoutput` or some other primitive is defined is not a reliable way to determine if PDF output is being generated. We made this backward compatible as best we could this year, but next year, `\pdfoutput` may be defined even when DVI is being written.

- pdfTEX (http://pdftex.org) has many new features:

  - `\pdfmapfile` and `\pdfmapline` provide font map support from within a document.
  - Microtypographic font expansion can be used more easily. http://www.ntg.nl/pipermail/ntg-pdftex/2004-May/000504.html
  - All parameters previously set through the special configuration file `pdftex.cfg` must now be set through primitives, typically in `pdftexconfig.tex`; `pdftex.cfg` is no longer supported. Any extant `.fmt` files must be redumped when `pdftexconfig.tex` is changed.
  - See the pdfTEX manual for more: `texmf-dist/doc/pdftex/manual/pdftex-a.pdf`.

- The `\input` primitive in tex (and mf and mpost) now accepts double quotes containing spaces and other special characters. Typical examples:

```
\input "filename with spaces"   % plain
\input{"filename with spaces"}  % latex
```

  See the Web2C manual for more: `texmf-dist/doc/web2c`.

- encTEX support is now included within Web2C and consequently all TEX programs, via the `-enc` option — *only when formats are built*. encTEX supports general re-encoding of input and output, enabling full support of Unicode (in UTF-8). See `texmf-dist/doc/generic/enctex/` and http://www.olsak.net/enctex.html.

- Aleph, a new engine combining ε-TEX and Ω, is available. A little information is available in `texmf-dist/doc/aleph/base` and http://www.tex.ac.uk/cgi-bin/texfaq2html?label=aleph. The LATEX-based format for Aleph is named lamed.

- The latest LATEX release has a new version of the LPPL — now officially a Debian-approved license. Assorted other updates, see the `ltnews` files in `texmf-dist/doc/latex/base`.

- dvipng, a new program for converting DVI to PNG image files, is included. See http://www.ctan.org/pkg/dvipng.

- We reduced the cbgreek package to a "medium" sized set of fonts, with the assent and advice of the author (Claudio Beccari). The excised fonts are the invisible, outline, and transparency ones, which are relatively rarely used, and we needed the space. The full set is of course available from CTAN (http://mirror.ctan.org/tex-archive/fonts/greek/cbfonts).

- oxdvi has been removed; just use xdvi.

- The ini and vir commands (links) for tex, mf, and mpost are no longer created, such as initex. The ini functionality has been available through the command-line option `-ini` for years now.

- i386-openbsd platform support was removed. Since the tetex package in the BSD Ports system is available, and GNU/Linux and FreeBSD binaries were available, it seemed volunteer time could be better spent elsewhere.

- On sparc-solaris (at least), you may have to set the LD_LIBRARY_PATH environment variable to run the t1utils programs. This is because they are compiled with C++, and there is no standard location for the runtime libraries. (This is not new in 2004, but wasn't previously documented.) Similarly, on mips-irix, the MIPSpro 7.4 runtimes are required.

9.1.3 2005

2005 saw the usual huge number of updates to packages and programs. The infrastructure stayed relatively stable from 2004, but inevitably there were some changes there as well:

- New scripts texconfig-sys, updmap-sys, and fmtutil-sys were introduced, which modify the configuration in the system trees. The texconfig, updmap, and fmtutil scripts now modify user-specific files, under $HOME/.texlive2005.
- Corresponding new variables TEXMFCONFIG and TEXMFSYSCONFIG to specify the trees where configuration files (user or system, respectively) are found. Thus, you may need to move personal versions of fmtutil.cnf and updmap.cfg to these places; another option is to redefine TEXMFCONFIG or TEXMFSYSCONFIG in texmf.cnf. In any case the real location of these files and the values of TEXMFCONFIG and TEXMFSYSCONFIG must agree. See section 2.3, p. 5.
- Last year, we kept \pdfoutput and other primitives undefined for DVI output, even though the pdfetex program was being used. This year, as promised, we undid that compatibility measure. So if your document uses \ifx\pdfoutput\undefined to test if PDF is being output, it will need to be changed. You can use the package ifpdf.sty (which works under both plain TeX and LaTeX) to do this, or steal its logic.
- Last year, we changed most formats to output (8-bit) characters as themselves (see previous section). The new TCX file empty.tcx now provides an easier way to get the original ^^ notation if you so desire, as in:

  ```
  latex --translate-file=empty.tcx yourfile.tex
  ```

- The new program dvipdfmx is included for translation of DVI to PDF; this is an actively maintained update of dvipdfm (which is also still available for now, though no longer recommended).
- The new programs pdfopen and pdfclose are included to allow reloading of pdf files in the Adobe Acrobat Reader without restarting the program. (Other pdf readers, notably xpdf, gv, and gsview, have never suffered from this problem.)
- For consistency, the variables HOMETEXMF and VARTEXMF have been renamed to TEXMFHOME and TEXMFSYSVAR, respectively. There is also TEXMFVAR, which is by default user-specific. See the first point above.

9.1.4 2006–2007

In 2006–2007, the major new addition to TeX Live was the XeTeX program, available as the xetex and xelatex programs; see http://scripts.sil.org/xetex.

MetaPost also received a notable update, with more planned for the future (http://tug.org/metapost/articles), likewise pdfTeX (http://tug.org/applications/pdftex).

The TeX .fmt (high-speed format) and the similar files for MetaPost and METAFONT are now stored in subdirectories of texmf/web2c, instead of in the directory itself (although the directory is still searched, for the sake of existing .fmt's). The subdirectories are named for the 'engine' in use, such as tex or pdftex or xetex. This change should be invisible in normal use.

The (plain) tex program no longer reads %& first lines to determine what format to run; it is the pure Knuthian TeX. (LaTeX and everything else do still read %& lines).

Of course the year also saw (the usual) hundreds of other updates to packages and programs. As usual, please check CTAN (http://mirror.ctan.org) for updates.

Internally, the source tree is now stored in Subversion, with a standard web interface for viewing the tree, as linked from our home page. Although not visible in the final distribution, we expect this will provide a stable development foundation for future years.

Finally, in May 2006 Thomas Esser announced that he would no longer be updating teTeX (http://tug.org/tetex). As a result, there was been a surge of interest in TeX Live, especially among

GNU/Linux distributors. (There is a new `tetex` installation scheme in TeX Live, which provides an approximate equivalent.) We hope this will eventually translate to improvements in the TeX environment for everyone.

9.1.5 2008

In 2008, the entire TeX Live infrastructure was redesigned and reimplemented. Complete information about an installation is now stored in a plain text file `tlpkg/texlive.tlpdb`.

Among other things, this finally makes possible upgrading a TeX Live installation over the Internet after the initial installation, a feature MiKTeX has provided for many years. We expect to regularly update new packages as they are released to CTAN.

The major new engine LuaTeX (http://luatex.org) is included; besides a new level of flexibility in typesetting, this provides an excellent scripting language for use both inside and outside of TeX documents.

Support among Windows and the Unix-based platforms is now much more uniform. In particular, most Perl and Lua scripts are now available on Windows, using the Perl internally distributed with TeX Live.

The new tlmgr script (section 5) is the general interface for managing TeX Live after the initial installation. It handles package updates and consequent regeneration of formats, map files, and language files, optionally including local additions.

With the advent of tlmgr, the texconfig actions to edit the format and hyphenation configuration files are now disabled.

The xindy indexing program (http://xindy.sourceforge.net/) is now included on most platforms.

The kpsewhich tool can now report all matches for a given file (option `-all`) and limit matches to a given subdirectory (option `-subdir`).

The dvipdfmx program now includes functionality to extract bounding box information, via the command name extractbb; this was one of the last features provided by dvipdfm not in dvipdfmx.

The font aliases `Times-Roman`, `Helvetica`, and so on have been removed. Different packages expected them to behave differently (in particular, to have different encodings), and there was no good way to resolve this.

The platex format has been removed, to resolve a name conflict with a completely different Japanese platex; the polski package is now the main Polish support.

Internally, the `WEB` string pool files are now compiled into the binaries, to ease upgrades.

Finally, the changes made by Donald Knuth in his 'TeX tuneup of 2008' are included in this release. See http://tug.org/TUGboat/Articles/tb29-2/tb92knut.pdf.

9.1.6 2009

In 2009, the default output format for Lua(LaTeX is now PDF, to take advantage of LuaTeX's OpenType support, et al. New executables named `dviluatex` and `dvilualatex` run LuaTeX with DVI output. The LuaTeX home page is http://luatex.org.

The original Omega engine and Lambda format have been excised, after discussions with the Omega authors. The updated Aleph and Lamed remain, as do the Omega utilities.

A new release of the AMS Type 1 fonts is included, including Computer Modern: a few shape changes made over the years by Knuth in the Metafont sources have been integrated, and the hinting has been updated. The Euler fonts have been thoroughly reshaped by Hermann Zapf (see http://tug.org/TUGboat/Articles/tb29-2/tb92hagen-euler.pdf). In all cases, the metrics remain unchanged. The AMS fonts home page is http://www.ams.org/tex/amsfonts.html.

The new GUI front end TeXworks is included for Windows, and also in MacTeX. For other platforms, and more information, see the TeXworks home page, http://tug.org/texworks. It is a cross-platform front end inspired by the Mac OS X TeXShop editor, aiming at ease-of-use.

The graphics program Asymptote is included for several platforms. This implements a text-based graphics description language vaguely akin to MetaPost, but with advanced 3D support and other features. Its home page is http://asymptote.sourceforge.net.

The separate `dvipdfm` program has been replaced by `dvipdfmx`, which operates in a special compatibility mode under that name. `dvipdfmx` includes CJK support and has accumulated many other fixes over the years since the last `dvipdfm` release. The DVIPDFMx home page is http://project.ktug.or.kr/dvipdfmx.

Executables for the cygwin and i386-netbsd platforms are now included, while we were advised that OpenBSD users get TEX through their package systems, plus there were difficulties in making binaries that have a chance of working on more than one version.

A miscellany of smaller changes: we now use xz compression, the stable replacement for lzma (http://tukaani.org/xz/); a literal $ is allowed in filenames when it does not introduce a known variable name; the Kpathsea library is now multi-threaded (made use of in MetaPost); the entire TEX Live build is now based on Automake.

Final note on the past: all releases of TEX Live, along with ancillary material such as CD labels, are available at ftp://tug.org/historic/systems/texlive.

9.1.7 2010

In 2010, the default version for PDF output is now 1.5, enabling more compression. This applies to all the TEX engines when used to produce PDF and to `dvipdfmx`. Loading the pdf14 LATEX package changes back to PDF 1.4, or set `\pdfminorversion=4`.

pdf(LA)TEX now *automatically* converts a requested Encapsulated PostScript (EPS) file to PDF, via the epstopdf package, when and if the LATEX `graphics.cfg` configuration file is loaded, and PDF is being output. The default options are intended to eliminate any chance of hand-created PDF files being overwritten, but you can also prevent `epstopdf` from being loaded at all by putting `\newcommand{\DoNotLoadEpstopdf}{}` (or `\def...`) before the `\documentclass` declaration. It is also not loaded if the pst-pdf package is used. For more details, see the epstopdf package documentation (http://ctan.org/pkg/epstopdf-pkg).

A related change is that execution of a very few external commands from TEX, via the `\write18` feature, is now enabled by default. These are commands are `repstopdf`, `makeindex`, `kpsewhich`, `bibtex`, and `bibtex8`; the list is defined in `texmf.cnf`. Environments which must disallow all such external commands can deselect this option in the installer (see section 3.2.4), or override the value after installation by running `tlmgr conf texmf shell_escape 0`.

Yet another related change is that BIBTEX and Makeindex now refuse to write their output files to an arbitrary directory (like TEX itself), by default. This is so they can now be enabled for use by the restricted `\write18`. To change this, the `TEXMFOUTPUT` environment variable can be set, or the `openout_any` setting changed.

XeTEX now supports margin kerning along the same lines as pdfTEX. (Font expansion is not presently supported.)

By default, tlmgr now saves one backup of each package updated (`tlmgr option autobackup 1`), so broken packages updates can be easily reverted with `tlmgr restore`. If you do post-install updates, and don't have the disk space for the backups, run `tlmgr option autobackup 0`.

New programs included: the pTEX engine and related utilities for typesetting Japanese; the BIBTEXU program for Unicode-enabled BIBTEX; the chktex utility (http://baruch.ev-en.org/proj/chktex) for checking (LA)TEX documents; the dvisvgm (http://dvisvgm.sourceforge.net) DVI-to-SVG translator.

Executables for these new platforms are now included: `amd64-freebsd`, `amd64-kfreebsd`, `i386-freebsd`, `i386-kfreebsd`, `x86_64-darwin`, `x86_64-solaris`.

A change in TEX Live 2009 that we failed to note: numerous TEX4ht-related executables (http://tug.org/tex4ht) were removed from the binary directories. The generic `mk4ht` program can be used to run any of the various `tex4ht` combinations.

Finally, the TEX Live release on the TEX Collection DVD can no longer be run live (oddly enough). A single DVD no longer has enough room. One beneficial side effect is that installation from the physical DVD is much faster.

9.1.8 2011

The Mac OS X binaries (`universal-darwin` and `x86_64-darwin`) now work only on Leopard or later; Panther and Tiger are no longer supported.

The `biber` program for bibliography processing is included on common platforms. Its development is closely coupled with the `biblatex` package, which completely reimplements the bibliographical facilities provided by LaTeX.

The MetaPost (`mpost`) program no longer creates or uses `.mem` files. The needed files, such as `plain.mp`, are simply read on every run. This is related to supporting MetaPost as a library, which is another significant though not user-visible change.

The updmap implementation in Perl, previously used only on Windows, has been revamped and is now used on all platforms. There shouldn't be any user-visible changes as a result, except that it runs much faster.

The initex and inimf programs were restored (but no other ini* variants).

9.1.9 2012

tlmgr supports updates from multiple network repositories. The section on multiple repositories in the tlmgr help output has more.

The parameter \XeTeXdashbreakstate is set to 1 by default, for both xetex and xelatex. This allows line breaks after em-dashes and en-dashes, which has always been the behavior of plain TEX, LATEX, LuaTEX, etc. Existing XeTEX documents which must retain perfect line-break compatibility will need to set \XeTeXdashbreakstate to 0 explicitly.

The output files generated by pdftex and dvips, among others, can now exceed 2gb.

The 35 standard PostScript fonts are included in the output of dvips by default, since so many different versions of them are extant.

In the restricted \write18 execution mode, set by default, mpost is now an allowed program.

A texmf.cnf file is also found in ../texmf-local, e.g., /usr/local/texlive/texmf-local/web2c/texmf.cnf, if it exists.

The updmap script reads a per-tree updmap.cfg instead of one global config. This change should be invisible, unless you edited your updmap.cfg's directly. The updmap --help output has more.

Platforms: armel-linux and mipsel-linux added; sparc-linux and i386-netbsd are no longer in the main distribution.

9.1.10 2013

Distribution layout: the top-level texmf/ directory has been merged into texmf-dist/, for simplicity. Both the TEXMFMAIN and TEXMFDIST Kpathsea variables now point to texmf-dist.

Many small language collections have been merged together, to simplify installation.

MetaPost: native support for PNG output and floating-point (IEEE double) has been added.

LuaTEX: updated to Lua 5.2, and includes a new library (pdfscanner) to process external PDF page content, among much else (see its web pages).

XeTEX (also see its web pages for more):

- The HarfBuzz library now used for font layout instead of ICU. (ICU is still used to support input encodings, bidirectionality, and the optional Unicode line breaking.)
- Graphite2 and HarfBuzz are used instead of SilGraphite for Graphite layout.
- On Macs, Core Text is used instead of the (deprecated) ATSUI.
- Prefer TrueType/OpenType fonts to Type1 when the names are the same.
- Fix occasional mismatch in font finding between XeTEX and xdvipdfmx.
- Support OpenType math cut-ins.

xdvi: now uses FreeType instead of t1lib for rendering.

microtype.sty: some support for XeTEX (protrusion) and LuaTEX (protrusion, font expansion, tracking), among other enhancements.

tlmgr: new pinning action to ease configuring multiple repositories; that section in tlmgr --help has more, online at http://tug.org/texlive/doc/tlmgr.html#MULTIPLE-REPOSITORIES.

Platforms: armhf-linux, mips-irix, i386-netbsd, and amd64-netbsd added or revived; powerpc-aix removed.

9.1.11 2014

2014 saw another TEX tune-up from Knuth; this affected all engines, but the only visible change likely is the restoration of the preloaded format string on the banner line. Per Knuth, this now reflects the format that *would* be loaded by default, rather than an undumped format that is actually preloaded in the binary; it may be overridden in various ways.

pdfTeX: new warning-suppression parameter `\pdfsuppresswarningpagegroup`; new primitives for fake interword spaces to help with PDF text reflowing: `\pdfinterwordspaceon`, `\pdfinterwordspaceoff`, `\pdffakespace`.

LuaTeX: Notable changes and fixes were made to font loading and hyphenation. The biggest addition is a new engine variant, `luajittex` (http://foundry.supelec.fr/projects/luajittex) and its siblings `texluajit` and `texluajitc`. This uses a just-in-time Lua compiler (detailed *TUGboat* article at http://tug.org/TUGboat/tb34-1/tb106scarso.pdf). `luajittex` is still in development, is not available on all platforms, and is considerably less stable than `luatex`. Neither we nor its developers recommend using it except for the specific purpose of experimenting with jit on Lua code.

XeTeX: The same image formats are now supported on all platforms (including Mac); avoid Unicode compatibility decomposition fallback (but not other variants); prefer OpenType to Graphite fonts, for compatibility with previous XeTeX versions.

MetaPost: A new numbersystem `decimal` is supported, along with a companion internal `numberprecision`; a new definition of `drawdot` in `plain.mp`, per Knuth; bug fixes in SVG and PNG output, among others.

The pstopdf ConTeXt utility will be removed as a standalone command at some point after the release, due to conflicts with OS utilities of the same name. It can still (and now) be invoked as `mtxrun -script pstopdf`.

psutils has been substantially revised by a new maintainer. As a result, several seldom-used utilities (`fix*`, `getafm`, `psmerge`, `showchar`) are now only in the `scripts/` directory rather than being user-level executables (this can be reversed if it turns out to be problematic). A new script, `psjoin`, has been added.

The MacTeX redistribution of TeX Live (section 3.1.2) no longer includes the optional Mac-only packages for the Latin Modern and TeX Gyre fonts, since it is easy enough for individual users to make them available to the system. The convert program from ImageMagick has also been excised, since TeX4ht (specifically `tex4ht.env`) now uses Ghostscript directly.

The langcjk collection for Chinese, Japanese, and Korean support has been split into individual language collections for the sake of more moderate sizes.

Platforms: x86_64-cygwin added, mips-irix removed; Microsoft no longer supports Windows XP, so our programs may start failing there at any time.

9.2 Present—2015

LaTeX 2ε now incorporates, by default, changes previously included only by explicitly loading the fixltx2e package, which is now a no-op. A new latexrelease package and other mechanisms allow for controlling what is done. The included LaTeX News #22 and "LaTeX changes" documents have details. Incidentally, the babel and psnfss packages, while core parts of LaTeX, are maintained separately and are not affected by these changes (and should still work).

Internally, LaTeX 2ε now includes Unicode-related engine configuration (what characters are letters, naming of primitives, etc.) which was previously part of TeX Live. This change is intended to be invisible to users; a few low-level internal control sequences have been renamed or removed, but the behavior should be just the same.

pdfTeX: Support JPEG Exif as well as JFIF; do not even emit a warning if `\pdfinclusionerrorlevel` is negative; sync with xpdf 3.04.

LuaTeX: New library newtokenlib for scanning tokens; bug fixes in the `normal` random number generator and other places.

XeTeX: Image handling fixes; xdvipdfmx binary looked for first as a sibling to xetex; internal `XDV` opcodes changed.

MetaPost: New numbersystem `binary`; new Japanese-enabled upmpost and updvitomp programs, analogous to up*tex.

MacTeX: Updates to the included Ghostscript package for CJK support. The TeX Distribution Preference Pane now works in Yosemite (Mac OS X 10.10). Resource-fork font suitcases (generally without an extension) are no longer supported by XeTeX; data-fork suitcases (`.dfont`) remain supported.

Infrastructure: The fmtutil script has been reimplemented to read `fmtutil.cnf` on a per-tree basis, analogous to updmap. Web2C mktex* scripts (including mktexlsr, mktextfm, mktexpk) now prefer programs in their own directory, instead of always using the existing `PATH`.

Platforms: *-kfreebsd removed, since TeX Live is now easily available through the system platform mechanisms. Support for some additional platforms is available as custom binaries (http://tug.org/texlive/custom-bin.html). In addition, some platforms are now omitted from the DVD (simply to save space), but can be installed normally over the net.

9.3 Future

TEX Live is not perfect! (And never will be.) We intend to continue to release new versions, and would like to provide more help material, more utilities, more installation programs, and (of course) an ever-improved and better-checked tree of macros and fonts. This work is all done by volunteers in their spare time, and so there is always more to do. Please see http://tug.org/texlive/contribute.html.

Please send corrections, suggestions, and offers of help to:

tex-live@tug.org
http://tug.org/texlive

Happy TEXing!