
Installing T_EX Live 2010 on Ubuntu

Enrico Gregorio

Abstract

Is it possible to have T_EX Live 2010 installed along with the Ubuntu version of the GNU/Linux operating system? Yes, and here's how it may be done.

1 Introduction

A major drawback of the T_EX packaging on Ubuntu systems is that, due to a specific choice by the Debian developers, the maintenance program `tlmgr` is missing. On the one hand this protects the user from possible damages to the system; on the other hand, it prevents dynamic updates of the T_EX distribution, in order to get bug fixes or to have available all the new functions that ship almost every day. Moreover, as of November 2010, the T_EX Live provided by Ubuntu (by Debian, one should say) is still the 2009 edition, frozen as of October 2009.

We describe here a procedure to install T_EX Live 2010 on a desktop Ubuntu system; this T_EX Live will have its `tlmgr` for maintenance and updates. And, most important, it will not put anything dangerous in system directories. Only two (or three) files will be stored under `/etc`, in subdirectories already reserved for system customization.

To revert to using the system T_EX, the entire T_EX Live installation may be deleted from the system by a simple

```
$ rm -fr /usr/local/texlive/2010
```

and optionally by deleting the added files under `/etc`.

The procedure we'll describe can be adapted, with suitable changes, also to other distributions such as Fedora and OpenSUSE. Other Debian-based distributions should work the same as Ubuntu. We chose Ubuntu because it seems a frequently used distribution found on desktop GNU/Linux systems.

An Italian version of this paper appeared also on *ArsT_EXnica*. My hope is that this article will become an installation program: it shouldn't be very difficult for a Unix guru to turn this procedure into a set of scripts for the various distributions.

2 A short introduction to the terminal

The procedure will require a certain familiarity with the terminal, that is, the interface to the command line. Anyone who hasn't the slightest idea of what the terminal is should stop reading here. However, it's not so difficult to copy the commands exactly as they are written here. There are many introductions to the command line, for example

<https://help.ubuntu.com/community/UsingTheTerminal>

In what follows, a line such as

```
$ ls -l
```

denotes a command that must be typed on the terminal, followed by a carriage return in order to execute it. The symbol `$` represents the system prompt, on your terminal it may be different, for example something like

```
enrico@ubuntu:~$
```

Normally, after these initial characters, there's a flashing box. Copy the commands starting after the space following the `$`. Other `$` characters that are not at the start of the line must be typed. Sometimes the commands will be too long to be printed on one line in this article; they will be rendered with

```
$ command a b c \
d e f
```

The backslash denotes that the command continues on the following line. The spaces (or lack thereof) before the backslash are significant.

Answers by the system will be represented without the `$` symbol and underlined, for instance

```
bash: tix: command not found
```

says that the system would have executed the command `tix`, but it doesn't exist. The prefix `bash:` denotes the *shell* that's trying to execute the commands; ignore these details.

Almost always it's not necessary to copy completely the various pieces of a command line: press the TAB key and, if the completion is unique, the system will do it by itself.

Final advice: if your keyboard doesn't have a `~` key, find a way to type this character — and get, as soon as possible, a keyboard which has it.

► Notes for experienced users are introduced by this triangle. This paper assumes that the default shell is `bash`. Whoever uses a different shell qualifies as 'experienced user', and should be able to adapt the commands presented here to their particular shell(s).

3 Preliminaries

Install Perl-Tk and Perl-doc with Synaptic. Then open a terminal session and prepare a work directory, for example,

```
$ mkdir ~/texlive-install
$ cd ~/texlive-install
```

The latter command will put you inside the directory created with the former.

Now we'll briefly describe the two main ways to obtain T_EX Live: the first works completely on-line, the second can also be done off-line.

4 Obtaining the distribution (on-line)

The simplest way to install T_EX Live is through the Internet. Download the needed compressed archive by typing on the terminal

```
$ wget http://mirror.ctan.org/systems/\
texlive/tlnet/install-tl-unx.tar.gz
```

Now you have to uncompress the downloaded file. The command is

```
$ tar xzf install-tl-unx.tar.gz
```

which will produce a new directory, where we'll go:

```
$ cd install-tl-20100914
```

The final part of the name is the date when the installation program has been produced, so it can be different: the completion feature mentioned before is what we need to find out the right suffix. Go to section 6.

5 Obtaining the distribution (off-line)

If your link to the Internet does not suffice, you can download an ISO image of the distribution, which is a file equivalent to a DVD. The Web address to contact is

```
http://mirror.ctan.org/systems/texlive/\
Images/texlive2010.iso
```

It's a 1.9 GiB file; transfer it on a USB pen drive and copy it onto the machine where the installation is desired, or transform the ISO image to a DVD and insert it in the machine.

There is also a 'torrent': go to <http://www.tug.org/texlive/acquire-iso.html> and find the link to click; this should start the Transmission program that will download the image.

You can also get a physical DVD, either by joining a T_EX user group or purchasing one separately: <http://www.tug.org/texlive/acquire-dvd.html>.

Ultimately, a double click on the ISO image allows access to the virtual (or physical) disk. Open a terminal session, create a work directory with

```
$ mkdir ~/texlive-install
```

and go to the `texlive` directory on the virtual (or physical) disk with something like

```
$ cd /cdrom/TeXLive/texlive
```

Instead of "TeXLive" it might be something else; use the automatic completion. Go to section 6.

6 Installing the distribution

Now you have to type the installation command:

```
$ sudo ./install-tl -gui \
-repository http://mirror.ctan.org/\
systems/texlive/tlnet
```

The system will ask for an administrator password and a window similar to that in figure 1 will pop up.



Figure 1: Installation window

If your default paper size is Letter, press the corresponding 'Toggle' button (the default for T_EX Live is A4). At the bottom of the window, there's the **Install TeX Live** button. Press it and be confident that the installation will arrive at completion.

Do not modify the default setting for "Create symlinks in system directories": it *must* remain set to "No".

When the installation is finished, go to section 8.

7 If something goes wrong

If the installation doesn't finish, before retrying you must delete everything has been already written on your system:

```
$ cd /usr/local/texlive
$ ls
2010          texmf-local
$ sudo rm -rf 2010
$ cd -
```

Then retry. Pay attention because the command `rm` is very dangerous, so carefully copy the commands. The output from the second command (`ls`) should be as shown, except possibly `texmf-local` may be missing. If the answer is different, check the first command and retype it correctly.

8 Touching up the installation

Now comes the difficult part: make the system know where to find the T_EX distribution programs. Don't touch the standard distribution on Ubuntu, so you'll have no problem in installing programs that depend on it, such as Kile. There is some escape from this, but it requires installing the `equivs` package with `Synaptic` and doing some tricks that are very system dependent.

Go back to your work directory by

```
$ cd ~/texlive-install
```

and let's face up to the operating system. Give the mysterious commands

```
$ echo 'export PATH=/opt/texbin:${PATH}' \
> texlive.sh
$ sudo cp texlive.sh /etc/profile.d/
$ sudo mkdir -p /opt
```

This creates a file `texlive.sh` containing the text we have written between single quotes and copies it into a system directory. The decisive step now requires a choice which depends on the hardware architecture of your machine; give one, and only one, of the following commands

```
$ sudo ln -s /usr/local/texlive/2010/bin/\
i386-linux /opt/texbin
$ sudo ln -s /usr/local/texlive/2010/bin/\
x86_64-linux /opt/texbin
$ sudo ln -s /usr/local/texlive/2010/bin/\
powerpc-linux /opt/texbin
```

The author can't know which one: you must select the appropriate choice, depending on whether your machine is based on a 32-bit Intel (or AMD) processor, on a 64-bit x86 processor, or on a PowerPC chip. You can discover this by typing the command

```
$ ls /usr/local/texlive/2010/bin
```

which will answer with the required string.

► The experienced user might ask why we don't put inside `texlive.sh` the name of the directory with the executable programs, for example

```
export PATH=/usr/local/texlive/2010/\
bin/i386-linux:${PATH}
```

The idea is that when the TeX Live 2011 ships out, it will be sufficient to type the command

```
$ sudo ln -s /usr/local/texlive/2011/bin/\
i386-linux /opt/texbin
```

after installation, *without any other intervention*. Almost ... see later.

Now do a *logout*, because the system must digest the change. Then *login* again, open a terminal session and check that everything is ok; the command

```
$ which tex
```

should answer with

```
/opt/texbin/tex
```

If it's so, we are ready to update the distribution. Otherwise seek help from a guru.

There are two ways to use `tlmgr`; it's best to try both. The first starts `tlmgr` from the terminal. Type the command

```
$ gedit ~/.bashrc
```

and, in the window that opens, add, at the end,

```
# .bashrc addition for TeX Live
function sutlmgr () {
  if [[ -z "$@" ]]
  then
    sudo /opt/texbin/tlmgr -gui
  else
    sudo /opt/texbin/tlmgr "$@"
  fi
}
alias mktexlsr=
'sudo /opt/texbin/mktexlsr'
alias updmap-sys=\
'sudo /opt/texbin/updmap-sys'
alias fmtutil-sys=\
'sudo /opt/texbin/fmtutil-sys'
```

The three dots represent what is already in the file `.bashrc`, which must not be modified.

► The experienced user might prefer to make the change in the file `.bash_aliases`.

Save the change with the proper menu entry, exit `gedit` and at the terminal type

```
$ . ~/.bashrc
$ sutlmgr
```

The `tlmgr` window will appear, after the system has asked for a password. From now on the `sutlmgr` command will start `tlmgr` with administrator privileges. Thus, a command such as

```
$ sutlmgr show --list xyz
```

directly executes the requested `tlmgr` action. With

```
$ texdoc tlmgr
```

you can access the documentation about `tlmgr`.

One may also create a small application on the desktop (this holds for the Gnome window manager; there may be something similar for KDE). Point the mouse on the desktop, press the right button and choose "Create Launcher ...". In the window that will show up, write 'TeX Live Manager' in the 'Name' box, and in the 'Command' box type

```
gksu -d -S -D "TeX Live Manager"
'/opt/texbin/tlmgr -gui'
```

(all on one line, please). After creating the application a double click on its icon will ask for an administrator password and launch `tlmgr`.

There's a last thing to do: make the system aware of the OpenType fonts provided along with TeX Live, in order to use them with X_YTeX (L^A)TeX by font name and not just file name.

```
$ sudo cp \
$(kpsewhich -var-value TEXMFSYSVAR)\
/fonts/conf/texlive-fontconfig.conf \
/etc/fonts/conf.d/09-texlive.conf
$ sudo fc-cache -fsv
```

The commands for the procedure on a typical Ubuntu 10 installation are repeated in the final table 1; in italics are the parts that might be different; the underlined lines represent answers from the system, the one underlined and in italics represents what to type in place of `i386-linux` in the following line; the lines with actions written between angle brackets describe maneuvers that must be done outside the terminal.

9 OpenSUSE

The described procedure works with OpenSUSE, provided that Perl-Tk has been installed. There's no need to define a function to launch `tlmgr`, but you have to remember to use the options `-c` to `su` and `-E` to `sudo`. Thus call the maintenance programs as

```
$ su -c tlmgr -gui
$ sudo -E updmap-sys
$ sudo -E mktexlsr
```

10 Fedora

You can use the same procedure with Fedora, at least for version 13. The Perl-Tk module isn't included in the standard distribution. You can retrieve it from

```
http://koji.fedoraproject.org/koji/buildinfo?buildID=151517
```

The file `/etc/profile.d/texlive.sh` must be different and contain the following text:

```
#!/bin/bash
if ! echo $PATH | \
  /bin/egrep -q "(~|:)/opt/texbin($|:)"
then
  PATH=/opt/texbin:$PATH
fi
```

Another difference is that in Fedora only `root` is the administrator, so the commands that in Ubuntu are prefixed with `sudo` must be typed without it, after having executed an `su` command.

11 Updating to T_EX Live 2011

When the time comes, install the distribution as described in section 6. Then do

```
$ sudo ln -s /usr/local/texlive/2011/bin/\
i386-linux /opt/texbin
$ sudo cp \
$(kpsewhich -var-value TEXMFSYSVAR)\
/fonts/conf/texlive-fontconfig.conf \
/etc/fonts/conf.d/09-texlive.conf
$ sudo fc-cache -fsv
```

That's all (of course changing `i386-linux` if your architecture is different). No need to log out, no `texlive.sh` to write. Of course T_EX Live 2011 has not yet been released, but we are ready for it.

Appendix

In this appendix we assume that T_EX Live has been installed on Ubuntu, following the outlined procedure. Change the commands related to `tlmgr` if you run another distribution.

A Installing a personal package

Let's assume we need a L^AT_EX package that's not present in T_EX Live; this might happen due to licensing issues, or because we want to try an experimental version. There are two places where you can put the necessary files. First of all download the archive from where it is hosted (CTAN or somewhere else) and decompress it in a work directory. To give an example, the package will be `padua` and the directory will contain the files `README`, `padua.ins`, `padua.dtx`, and `padua.pdf` (the parts in italics represent the real name). Open a terminal session and type the following command:

```
$ tex padua.ins
```

If the file with extension `.ins` is not present, then

```
$ tex padua.dtx
```

will probably do the same job. In both cases the system will generate some files that we have to store in the correct place. Of course these are simple and generic instructions; some packages have a more complex structure and the author's directions should be followed, adapting them to what is found here.

Now you have to decide whether to use the *personal* tree or the *local* one. The main difference is that in the second case the package will be available to all users of the machine; of course you'll need administrator privileges to access the local tree.

The word 'tree' refers to a hierarchical directory structure, necessary to be accessed efficiently by the T_EX system.

On GNU/Linux systems, the personal tree is rooted at `~/texmf`, that is, it's a set of subdirectories of your home. The local tree is rooted at `/usr/local/texlive/texmf-local`. Actually it's not necessary to know where they are. We can define a shorthand to stand for the location of the chosen tree; for the personal tree the trick is to run

```
$ Local=$(kpsewhich -var-value TEXMFHOME)
```

while for the local tree the trick is

```
$ Local=$(kpsewhich -var-value TEXMFLOCAL)
```

The T_EX system is set up in such a way that it can know itself and the magic command is what stores in the variable `Local` the location of the chosen tree.

Let's now restart from where we stopped. We create the necessary directories and copy the files in them.

```
$ mkdir -p $Local/source/latex/padua
$ cp README padua.ins padua.dtx \
  $Local/source/latex/padua
$ mkdir -p $Local/doc/latex/padua
$ cp padua.pdf $Local/doc/latex/padua
$ mkdir -p $Local/tex/latex/padua
$ cp *.sty ... $Local/tex/latex/padua
```

That last line is intended to represent the files that should be seen by (L)TeX itself, which are typically generated from the dtx.

B Installing a font family

There are many instruction sets around the world about how to install new fonts we have bought or found for free. Installing fonts in the personal tree is not recommended, because it requires a constant labour by the user when updates to TeX Live include something related to fonts.

It's best to follow the instructions contained in the booklet "The font installation guide" by Philipp Lehman, available in TeX Live with the terminal command

```
$ texdoc fontinstallationguide
```

These instructions, by the author's choice, end with the preparation of the needed files. Let's assume that the font family is called '*Padua*', with TeX family name *zpd*. As usual the part to be changed will be in italics. Lehman's procedure creates a bunch of files in the work directory, with various extensions:

```
.tfm .vf .pfb .afm .map .sty .fd
```

These files must be stored in the correct place in the TeX system hierarchy. The correct place is the already described *local tree*. Again we don't need to know where it is; let's define a shorthand, create the infrastructure, and store the files.

```
$ Local=$(kpsewhich -var-value TEXMFLOCAL)
$ sudo mkdir -p \
  $Local/fonts/{afm,tfm,type1,vf}/padua
$ sudo cp zpd*.afm $Local/fonts/afm/padua
$ sudo cp zpd*.tfm $Local/fonts/tfm/padua
$ sudo cp \
  zpd*.pfb $Local/fonts/type1/padua
$ sudo cp zpd*.vf $Local/fonts/vf/padua
$ sudo mkdir -p $Local/tex/latex/padua
$ sudo cp *.sty *.fd $Local/tex/latex/padua
$ sudo mkdir -p \
  $Local/fonts/map/dvips/padua
$ sudo cp padua.map \
  $Local/fonts/map/dvips/padua
$ mktexlsr
```

We have placed the furniture, now we must provide the TeX system with the door key. There are two

cases: it's the first time we are installing a new font family, or we've already followed this procedure. In the first case we have to create a file and store it in a suitable place:

```
$ echo "Map padua.map" > updmap-local.cfg
$ mkdir -p $Local/web2c
$ sudo mv updmap-local.cfg $Local/web2c
$ sutlmgr generate --rebuild-sys updmap
```

In the second case, we only need to append a line to an already existing file:

```
$ cp $Local/web2c/updmap-local.cfg .
$ echo "Map padua.map" >> updmap-local.cfg
$ sudo mv updmap-local.cfg $Local/web2c
$ sutlmgr generate --rebuild-sys updmap
```

(The last action, as well as the call to `mktexlsr`, can be executed from the graphical interface of `tlmgr`.) By doing so we can be certain that the door key won't be lost with TeX Live updates. See also <http://www.tug.org/fonts/fontinstall.html> for a more verbose description of the steps.

If, by chance, also the OpenType versions of our new fonts are available, add the following couple of lines:

```
$ sudo mkdir -p $Local/fonts/opentype/padua
$ sudo cp *.otf $Local/fonts/opentype/padua
```

to the similar ones seen before. Similarly, by changing `opentype` with `truetype` if the font files have extension `.ttf`.

If it's the first time you add OpenType, TrueType or Type 1 fonts to the local tree, you have to make the new location known to the system. Do

```
$ cp /etc/fonts/conf.d/09-texlive.conf \
  09-texlive-local.conf
$ gedit 09-texlive-local.conf
```

In the window that will appear, change all the strings '2010/texmf-dist' into 'texmf-local'. Save and type in the terminal

```
$ sudo mv 09-texlive-local.conf \
  /etc/font/conf.d
$ sudo fc-cache -fsv
```

Now X(L)TeX will also be able to access the new fonts. Only the `fc-cache` call is needed if you already created the 'local' configuration file.

◇ Enrico Gregorio
Dipartimento di Informatica
Università di Verona
enrico dot gregorio (at) univr
dot it

Table 1: The whole procedure for Ubuntu

```

<Install Perl-Tk and Perl-doc with Synaptic>
<Start terminal session>
$ mkdir ~/texlive-install
$ cd ~/texlive-install
$ wget http://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
$ tar xzf install-tl-unx.tar.gz
$ cd install-tl-20100914
$ sudo ./install-tl -gui -repository http://mirror.ctan.org/systems/texlive/tlnet
<Press "Toggle" for the paper format if you need to>
<Press "Install TeX Live">
<Wait until the installation is complete; drink a coffee, maybe two>
<Press "End">
$ cd ~/texlive-install
$ echo 'export PATH=/opt/texbin:${PATH}' > texlive.sh
$ sudo cp texlive.sh /etc/profile.d/
$ sudo mkdir -p /opt
$ ls /usr/local/texlive/2010/bin
  i386-linux
$ sudo ln -s /usr/local/texlive/2010/bin/i386-linux /opt/texbin
<Do a logout>
<After login, open a terminal>
$ which tex
  /opt/texbin/tex
<If the answer is different, cry out 'Help'>
$ gedit ~/.bashrc
<Append to the file>
  # Additions for TeX Live
  function sutlmgr () {
    if [[ -z "$@" ]]
    then
      sudo /opt/texbin/tlmgr -gui
    else
      sudo /opt/texbin/tlmgr "$@"
    fi
  }
  alias mktexlsr='sudo /opt/texbin/mktexlsr'
  alias updmap-sys='sudo /opt/texbin/updmap-sys'
  alias fmtutil-sys='sudo /opt/texbin/fmtutil-sys'
<Save and exit from gedit>
$ sudo cp $(kpsewhich -var-value TEXMFSYSVAR)/fonts/conf/texlive-fontconfig.conf \
  /etc/fonts/conf.d/09-texlive.conf
$ sudo fc-cache -fsv
<Relax and enjoy TEX Live 2010>

Notes.
(1) The date 20100914 is an example, it will likely be different.
(2) i386-linux corresponds to one of the possible architectures (a.k.a. platforms);
it may also be x86_64-linux or, less probably, powerpc-linux.

```