The teTeX HOWTO: The Linux-teTeX Local Guide

Robert Kiesling v2.9.1, 21 August 1997

This document covers the basic installation and usage of the teTeX TeX and LaTeX implementation under the major U.S. Linux distributions, and auxiliary packages like Ghostscript. Please send all complaints, suggestions, errata, and any miscellany to kiesling@terracom.net, so I can keep this document as complete and up to date as possible.

Contents

1 Introduction. **2**

2 What is TeX? What is LaTeX? What is teTeX? **3**

2.1 Resources for further information. 4

3 Installation notes. **6**

3.1 Generic CTAN distribution, V. 0.4. 7

3.1.1 Installing the binary distribution. 7

3.1.2 Installing the source distribution. 9

3.2 Linux packages. 11

3.2.1 Slackware 3.2. 11

3.2.2 Debian GNU/Linux, V. 1.3. 12

3.2.3 RedHat V. 4.2. 13

3.3 Ghostscript V. 5.03. 13

3.4 APSFILTER. 14

3.5 The `lpd(8)` daemon. 15

4 Post-installation configuration details. **16**

4.1 What if my printer isn't included? 17

5 Using teTeX. **17**

5.1 Printing the documentation. 17

5.2 TeX and LaTeX commands. 19

5.2.1 Document structure. 19
5.2.2 Characters and type styles. 21
5.2.3 Margins and line spacing. 23
5.2.4 Document classes. 24
5.3 LaTeX extension packages and other resources. 27

6 Mixing text and graphics with `dvips(1)`. 28

7 Using PostScript fonts. 30

8 Appendix: CTAN Site Listing 32

1 Introduction.

FAQ No. 1. My computer just ate NINE high density diskettes' worth of data. WHAT HAPPENED?

Answer: Installing teTeX on Chanel3, my Compaq laptop, was like dropping a 20-foot concrete bridge section exactly into place from a height of 50 feet. teTeX is a *big package.* Even so, it is a moderately complete implementation of TeX 3.1415 and LaTeX 2e for Linux systems. TeX is a big subject anyway, so you can expect to spend the rest of your computing career keeping up-to-date on the latest in the world of TeX. That is to say, installing and using teTeX is not for the faint of heart. Nor is it for day trippers. This package requires serious quality time.

Thomas Esser, the author of teTeX, has gone to great lengths to make the package fast, complete, and easy to use. Because TeX is implemented for practically every serious computer system in the world—and quite a few "non-serious" ones—implementors must provide the installation facilities for all of them. This accounts in part for teTeX's size. It also accounts for the fact that the pieces necessary to make a workable teTeX installation are spread all over your friendly neighborhood CTAN archive.

CTAN is the Comprehensive TeX Archive Network, a series of anonymous FTP sites which archive TeX programs, macros, fonts, and documentation. You'll probably become familiar with at least one CTAN site. In this document, a pathname like `~CTAN/contrib/pstricks` means "look in the directory `contrib/pstricks` of your nearest CTAN site." See section 8 for a current list of CTAN sites and their mirror sites.

Fortunately, some considerate Linux Distribution implementors have assembled the necessary pieces for us. teTeX comes with all the major Linux distributions.

However, if you don't have the Slackware, RedHat, or Debian GNU/Linux distribution, you can install teTeX from its official CTAN distribution. In some cases this may be more desirable. See **Section 3** for details.

If you already have teTeX installed on your system and want to jump directly into figuring out how to use it, skip this section and the next, and go directly to section 5.

2 What is TeX? What is LaTeX? What is teTeX?

teTeX is an implementation of TeX for UNIX systems. It is the work of Thomas Esser, te@informatik.uni-hannover.de. In the Linux versions of teTeX, the executable programs themselves run under Linux and the fonts are provided in form usable by the Linux-teTeX system. (The sections covering teTeX installation concentrate on the i386 versions of Linux. Installing teTeX for MkLinux or Linux for the Alpha should require only substituting the appropriate binary-program archive in the installation process.) The rest of the code, TeX and LaTeX itself, is portable across various machines.

In addition to the executable programs, the distribution includes all of the TeX and LaTeX package, metafont and its sources, `bibtex(1), makeindex(1),` and *all* of the documentation... more than 4 megabytes' worth. The documentation covers everything you will forseeably need to know to get started. So, you should install all of the documents. Not only will you eventually read them, the documents themselves provide many examples of "live" TeX and LaTeX code.

In comparison with other implementations of TeX, the installation of teTeX is almost trivial, even without the Linux distribution packages, if you don't count the effort necessary acquire the distributions via anonymous FTP or insert and remove several dozen distribution diskettes by hand. If your teTeX distribution arrived on a CD-ROM, even less effort is required to install it.

TeX is a typesetting system developed by Professor Donald Knuth of Stanford University. It is a lower-level typesetting language that powers all of the higher-level packages like LaTeX. Essentially, LaTeX is a set of TeX macros which provide convenient, predefined document formats for end users. If you like the formats provided by LaTeX, you may never need to learn bare-bones TeX programming. The difference between the two languages is like the difference between assembly language and C. You can have the speed and flexibility of TeX, or the convenience of LaTeX. Which brings us to the next answer,

Answer: You have it backwards! I want to know what exactly I need to get before I can have TeX on my system!

It's important to remember that TeX only handles the typesetting part of the document preparation. Generating output with TeX is like compiling source code into object code, which still needs to be linked. You prepare an input file with a text editor – what most people think of as "word processing" – and typeset the input file document with TeX to produce a device-independent output file, called a `.dvi` file.

You also need output drivers for your printer and video display. These output drivers translate TeX's `.dvi` output to display your typeset document on the screen or on paper. This software is collectively known as "dviware." For example, TeX itself only makes requests for fonts. It is up to the `.dvi` output translator to provide the actual font to the display device if necessary, regardless of whether it is the screen or a printer. This extra step may seem overly complicated, but the abstraction allows documents to display the same on different devices with no change to the original document.

In fact, much of TeX's, and therefore LaTeX's, complexity, arises from its implementation of various font systems, and the way these fonts are specified. A major improvement of LaTeX 2e over its predecessor was the way users specify fonts, the former New Font Selection Scheme. (See the sections 5.2.2 and 7.)

teTeX comes distributed with about a dozen standard fonts preloaded, which is enough to get you started. Also provided are the font metrics descriptions, in `.tfm` (TeX font metric) files. To generate the other fonts you will need, it is simply a matter of installing the `metafont` sources. teTeX's `.dvi` utilities will invoke `metafont` automatically and generate the Computer Modern fonts you need, on-the-fly.

By the way, the letters of the word “TeX” are Greek, tau-epsilon-chi. This is *not* a fraternity. Instead, it is the root of the Greek word, *techne,* which means art and/or science. “TeX” is not pronounced like the first syllable in "Texas.” The *chi* has no English equivalent, but TeX is generally pronounced so that it rhymes with “yecch,” to use Professor Knuth’s example from *The TeXBook* (see below). When writing, "TeX,” on character devices, always use the standard capitalization, or the `\TeX{}` macro in typesetting. This is how TeX is distinguished from other typesetting systems.

Speaking of typing, any of the editors which work under Linux— `nvi(1), jed(1), joe(1), jove(1), vi(1), vim(1), stevie(1), emacs(1),` microemacs—will work to prepare a TeX input file, as long as the editor reads and writes plain-vanilla ASCII text. My preference is `emacs(1),` the GNU version. There are several reasons for this:

- Emacs’ TeX and LaTeX modes obviate the need for a stand-alone TeX shell.
- Emacs can automatically insert TeX-style, “curly quotes,” as you type, rather than the "ASCII-vanilla" kind.
- Emacs has integrated support for `texinfo` and `makeinfo,` a hypertext documentation system.
- Emacs is widely supported. Version 19.34, for example, is included in the major U.S. Linux distributions.
- Emacs does everything except butter the toast in the morning.
- Emacs is free.

There’s a lot of software to assemble. In the meantime, you can start in “learning” TeX and LaTeX. Remember that teTeX and the font packages have been designed as two separate entities: The teTeX executable programs and shell scripts, as distributed with Linux, have been built specifically for the system, but the CM, DC, American Mathematical Society, or other font distributions work on many different platforms. While you are working on assembling the files, you can take a few breaks to locate some of the documentation you will need.

2.1 Resources for further information.

There are user manuals available both commercially and via the Internet. Judging by the number of mentions they receive in the Usenet `comp.text.tex` newsgroup, the most useful—and definitive—commercially available texts for beginners are:

LaTeX: A Document Preparation System, by Leslie Lamport, 272 pp. If you’re using LaTeX instead of plain TeX (highly recommended), this is the definitive reference.

If you must use plain TeX, *The TeXBook* by Donald Knuth, 483 pp., is the definitive reference. It is also necessary if you plan to do any serious class, package, or macro writing for LaTeX.

The LaTeX Companion, by Michel Goosens, Frank Mittelbach, and Alexander Samarin, 530 pp., is more advanced than the Lamport, above. If you are approaching TeX or LaTeX for the first time, you may feel lost reading this. (I was.) However, when you need to add extension packages, like PSNFSS (See the section titled, 7.), or `bibtex(1)`, a bibliography indexing program, this book is one of the most highly regarded on the market.

At your nearest CTAN site you can retrieve these documents for free:

The Not So Short Introduction to LaTeX2e, by Tobias Oetiker, Hubert Partl, Irene Hyna, and Elisabeth Schlegl, 69 pp. This wonderful document is located at `~CTAN/packages/TeX/info/lshort/*.`

You can get a PostScript or `.dvi` version of the document ready for printing, or the native LaTeX document. There is also a version available in German: `lkurz.*.` Make sure to read the `README` file before assembling!

A Gentle Introduction to TeX: A Manual for Self-Study, by Michael Doob, 91 pp. You can find this document at: `~CTAN:packages/TeX/info/gentle.tex.` Almost of necessity, this document covers less ground than its LaTeX counterpart, above. However, it will get you to the same place as the LaTeX manuals. If you must use plain TeX for your documents, this document clarifies many of the complexities of plain TeX and makes its use almost easy.

"IMPRINT: The Newsletter of Digital Typography," edited by Robert Kiesling. I realize that this is BLATANT and SHAMELESS self-promotion. But, you should know anyway, that IMPRINT is a free, ASCII-text newsletter which is available via e-mail. IMPRINT appears approximately monthly and covers a broad range of text processing and digital imaging topics, both beginning and advanced. Many of the items covered apply directly or indirectly to TeX'ing. The emphasis is on production of industry-standard typeset and printed material. To subscribe to IMPRINT, send a brief, human-readable message to me at imprint@macline.com.

The LaTeX Catalogue is a `bibtex(1)` database of available LaTeX packages, compiled and maintained by Graham Williams. It's included with teTeX, and the most recent version is available on the World Wide Web. Do you need a package that prints borders, or makes margin notes? You'll find that the package you need is listed here. *The LaTeX Catalogue* is located in your local teTeX library in the directory `teTeX/texmf/doc/Catalog,` and on the Web at http://cbr.dit.csiro.au/~gjw. See section 5.3 for further details about LaTeX packages.

Thomas Merz's Ghostscript Manual, which is the Ghostscript appendix of his book, *PostScript & Acrobat/PDF: Applications, Troubleshooting, and Cross-Platform Publishing.* It is available from the Ghostscript Home Page (see the section 3.3), or from Merz's home page, http://www.muc.de/~tm/.

There are, of course, other guides available to using TeX and LaTeX. They cover different aspects of these systems to varying degrees. The reference documents cited above, however, are the most comprehensive in scope that I have seen and are aimed at beginners (or near-beginners).

If the going gets especially tough, you can probably do a little extra shopping at Office Max, Office Depot, Staples, or your local stationer, and pick up several reams of three-hole punched, photocopy paper, two or three, three-inch binders, and some index tabs. When it comes time to print the documents, you'll need a place to keep them, and they seem to be more useful if they are kept on paper. This must be one of the stranger phenomena of technical documentation.

You will note, however, that the references mentioned above are hardware-independent. They won't tell you a thing about running teTeX specifically. Many of them, in fact, refer to some mythical "Local Guide."

This, and several of the documents that come bundled with teTeX, comprise the less-than-mythical Local Guide to installing and operating teTeX with Linux.

3 Installation notes.

All of the major Linux distributions include packaged versions of teTeX, and each distribution has its own idiosyncrasies. The packaging methods of each distribution are, for the most part, incompatible. If you try to install teTeX from another distribution, you *may* succeed in installing the package, but you're certain to mess up the package-management database on your system. When installing teTeX, please consult the section below that corresponds to your Linux distribution.

Installing teTeX is surprisingly easy for a package of this magnitude. This document covers only the major free U.S. Linux distributions, because I haven't had time to obtain or install European Linux distributions like S.u.S.E.

However, the generic, teTeX distribution isn't any harder to install than the Linux packages. See section 3.1, below.

You should consider installing the generic teTeX distribution from the CTAN archives if:

- Your system isn't based on one of the standard Linux distributions.
- You don't have root privileges on your system.
- You want or need to have the very latest version of teTeX, or LaTeX.
- You don't have enough disk space available for a full installation.
- You want to install teTeX somewhere instead of the `/usr` file system.
- You would like to share your teTeX installation with other UNIX variants or platforms on a network. In this case, you should strongly consider installing from the *source* distribution. See section 3.1.2, below.
- You want the latest versions of teTeX's public domain Type 1 fonts, which are significantly better than the fonts included in earlier releases.

A complete installation of the binary distribution requires 40-50 Mb of disk space, and building the distribution from the source code takes about 75 Mb, so you should make sure that the disk space is available before you start. You don't need to have the `gcc(1)` compiler or the X Windows System installed (although X certainly helps because it is much easier to preview documents on-screen). All you need is an editor that is capable of producing plain ASCII, text (see section 2). What could be simpler?

Ghostscript V. 5.03 allows printing of PostScript documents on non-PostScript printers, and allows previewing of PostScript documents on VGA monitors and X Window System displays. If you already have a PostScript printer, you won't need Ghostscript simply to print PostScript documents. Ghostscript has many other capabilities, however, which are beyond the scope of this HOWTO.

APSFILTER can automate document post processing and printing, and make life with your printer a lot easier. See the section titled 3.4.

For information on how to install a printer daemon and generally configure printers for Linux, see the section titled 3.5, and consult the Printing-HOWTO.

3.1 Generic CTAN distribution, V. 0.4.

You can retrieve the files from one of the CTAN archives listed in section 8. In the examples below, the files were retrieved from the CTAN archive at ftp.tex.ac.uk.

3.1.1 Installing the binary distribution.

Minimal installation. First, FTP to ftp.tex.ac.uk and `cd` to the directory

```
ctan/tex-archive/systems/unix/teTeX/distrib/
```

Retrieve the files

```
INSTALL.bin
install.sh
```

and place them in the top-level directory where you want to install teTeX, for example, `/var/teTeX` if you plan to install teTeX in the `/var` file system.

Print out the `INSTALL.bin` file. Keep this file handy, because it describes how to install a minimal teTeX installation. The minimal installation requires only 10-15 MB of disk space, but it is recommended that you install the complete teTeX package if at all possible. For a minimum installation, you'll need the files

```
ctan/tex-archive/systems/unix/teTeX/distrib/base/latex-base.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/base/tetex-base.tar.gz
```

You'll also need one of two archives which contain the executable teTeX programs. Retrieve the archive file

```
ctan/tex-archive/systems/unix/teTeX/distrib/binaries/i386-linux.tar.gz
```

if your system uses the Linux ELF shared libraries, `ld.so(1)` of at least version 1.73, and clibs of at least version 5.09. If it doesn't, retrieve the archive

```
ctan/tex-archive/systems/unix/teTeX/distrib/binaries/i386-linuxaout.tar.gz
```

which is compiled for systems that use the older, a.out-format static libraries.

Then, following the instructions in the file `INSTALL.bin`, execute the command

```
sh ./install.sh
```

while in the top-level teTeX installation directory. (Make sure that the teTeX archives are located there, too.) After a few moments, the installation program will warn you that you are missing some of the teTeX packages. However, if you're planning only a minimal teTeX installation, you should ignore the warnings and proceed. To configure the basic teTeX system, see section 3.1.1, below.

To install the remaining packages, see the next section.

Complete installation. To perform a complete teTeX installation, retrieve the archive files listed in the previous section, as well as the following files:

```
ctan/tex-archive/systems/unix/teTeX/distrib/doc/ams-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/bibtex-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/eplain-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/fonts-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/general-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/generic-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/latex-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/makeindex-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/metapost-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/doc/programs-doc.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/fonts/ams-fonts.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/fonts/dc-fonts.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/fonts/ec-fonts.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/fonts/misc-fonts.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/fonts/postscript-fonts.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/fonts/sauter-fonts.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/amstex.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/bibtex.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/eplain.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/latex-extra.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/metapost.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/pictex.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/pstricks.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/texdraw.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/goodies/xypic.tar.gz
```

All of these files should be placed in the top-level directory where you want teTeX to reside. As with the minimal installation, execute the command

```
sh ./install.sh
```

Base system configuration. The `install.sh` script, after determining which teTeX archive series are present, will present you with a menu of options. The only setting you need to make at this point is to set the top-level directory where you want teTeX installed, by selecting the "D" option. You must, of course, choose a directory in whose parent directory you have write permissions. For example, if you are installing teTeX in your home directory, you would specify the teTeX installation directory as

```
/home/john.q.public/teTeX
```

and, after returning to the main menu, select "I" to proceed with the installation. Note that the directory must not exist already: the `install.sh` script must be able to create it.

An option which you should consider enabling, is setting an alternative directory for generated fonts. Even if you plan to use only PostScript-format, Type 1 scalable fonts, occasionally you'll process a file that requires the Computer Modern fonts. Enabling this option requires that you enter the directory to use. You must have write permissions for the parent directory. Following the example above, you could specify

```
/home/john.q.public/texfonts
```

or, if you want the generated fonts to be accessible by all users on the system, specify a directory like

```
/var/texfonts
```

I would recommend that you *not,* however, use the default `/var/tmp/texfonts` directory for this option, because the generated fonts could be deleted after the next reboot, and the fonts will need to be generated again the next time they're needed.

After you've selected the option "I", and `install.sh` has installed the archives, set various permissions, and generated its links and format files, the program will exit with a message telling you to add the teTeX binary directory to your `$PATH` environment variable, and the directories where the man pages and info files reside to your `$MANPATH` and `$INFOPATH` environment variables. For example, add the statements

```
export PATH=$PATH:"/home/john.q.public/teTeX/bin"
export MANPATH=$MANPATH":/home/john.q.public/teTeX/man"
export INFOPATH$=INFOPATH":/home/john.q.public/teTeX/info"
```

to your `~/.bash_profile` if you use `bash(1)` as your shell, or to your `~/.profile` if you use another shell for logins.

Log out, and then log in again, so the environment variables are registered. Then, run the command

```
texconfig confall
```

to insure that the installation is correct.

Next, you can configure teTeX for you specific hardware. See section 4, below.

3.1.2 Installing the source distribution.

To install teTeX V. 0.4 from the source code, `ftp` to a CTAN site like ftp://ftp.tex.ac.uk and retrieve the files

```
ctan/tex-archive/systems/unix/teTeX/distrib/INSTALL.src
ctan/tex-archive/systems/unix/teTeX/distrib/sources/README.texmf-src
ctan/tex-archive/systems/unix/teTeX/distrib/sources/teTeX-lib-0.4pl8.tar.gz
ctan/tex-archive/systems/unix/teTeX/distrib/sources/teTeX-src-0.4pl7.tar.gz
```

Read over the instructions in `INSTALL.src`, then `su` to root and unpack the files in a directory for which you have read-write-execute permissions.

Remember to use the `p` argument to `tar(1)`, and also remember to unset the `noclobber` option of `bash(1)`. You can do this with the counterintuitive command

```
set +o noclobber
```

Note that the argument `+o` to `set` *un*sets a variable, just exactly backwards from what you might expect.

The file `teTeX-lib-0.4pl8.tar.gz` will create the directory `./teTeX`. The file `teTeX-src-0.4pl7.tar.gz` will create the directory `teTeX-src-0.4` Print out the file `INSTALL.src` and keep it nearby for the following steps. `cd` to the `./teTeX-src-0.4` directory, and, per the instructions in the `INSTALL.src` file, edit `./Makefile`. You need to set the `TETEXDIR` variable to the absolute path of the parent teTeX directory. This will be the subdirectory `teTeX` of the directory where you unpacked the source and library archives. For example, if you unpacked the archives in your home directory, you would set `TETEXDIR` to

```
/home/john.q.public/teTeX
```

The rest of the `Makefile` options are pretty generic. With `gcc(1)` version 2.7.2 and later, you should not need to make any further adjustments unless you have a non-standard compiler and library setup, or want the compiler to perform some further optimizations, or for some other reason. Check that the `USE_DIALOG`, `USE_NCURSES`, and `HAVE_NCURSES` variables are set correctly for your system, because the `dialog` program needs the ncurses library to be installed. A `ncurses(3x)` library is included in the source distribution, so the default values in the `Makefile` should work fine. If you can't get `ncurses(3x)` to compile or link, `texconfig(1)` can also be run from the command line.

If you've done everything correctly up to this point, you should be able to type `make world` in the top-level source directory, and relax until the teTeX executables are built. This can take a few hours.

After the build has completed, set the environment variables `$PATH, $MANPATH`, and `$INFOPATH` to include the teTeX directories. The statements which would be added to the file `~/.bash_profile`, in the example, above, would be

```
export PATH=$PATH":/home/john.q.public/teTeX/bin/i386-linux"
export MANPATH=$MANPATH":/home/john.q.public/teTeX/man"
export INFOPATH=$INFOPATH":/home/john.q.public/teTeX/info"
```

The `$PATH` variable is different in the source distribution than in the binary distribution. Note that here the path to the binaries is `teTeX/bin/i386-linux` instead of simply `teTeX/bin` as in the binary distribution.

At this point you can run `texconfig confall` to ensure that the paths have been set correctly, and then proceed to configure teTeX as in the binary distribution. See the section 4, below.

3.2 Linux packages.

3.2.1 Slackware 3.2.

First, `ftp` to your nearest Linux archive site. Mine is `wuarchive.wustl.edu.` Then find the directory with the Slackware distribution diskettes. On `wuarchive.wustl.edu`, this is

```
systems/linux/sunsite/distributions/Slackware/slakware/.
```

Linux sites which mirror `sunsite.unc.edu` will store these diskettes in the directory `distributions/Slackware/slakware/.` teTeX, the full package, is contained on the Slackware disk series `t.` So, grab all nine disks' worth of the `t` series, disks `t1 - t9.` Be sure to keep them in order, too. Either store the files in separate subdirectories labeled `t1 - t9` on a hard drive partition, or on diskettes, and label the diskettes `t1` through `t9.` We're going to install them by hand.

This isn't difficult. The Slackware installer creates the directories and unpacks the files. It also provides descriptions of each module in the distribution, which allows you to decide whether you want to install it or not. In the case of teTeX, however, you are simply going to install everything, because that's what you should do anyway.

Let's assume that you have all nine diskettes' worth of the Slackware teTeX distribution ready at hand, organized as described above. You'll have a lot of files which have the extension `.tgz.` This is shorthand for a `tar(1)` archive compressed with `gzip(1).` The names all fit the 8+3 filename limitations of MS-DOG. Aren't you glad you decided to scrap your DOG partitions and install Linux instead? You can use a MS-DOG hard disk partition or DOG-format diskettes to store the files. The archives also begin with the letters `tb, td,` or `tm,` and so on, which is the implementors' shorthand for TeX binary, TeX documentation, TeX macro, and so on. The difference to you is academic, because you'll be installing everything anyway.

Simply fire up the Slackware `install` utility. You needn't concern yourself with reconfiguring the system, so select the option to add new software. Select the appropriate source media (diskettes, HD partition, or CD-ROM), specify that you want to install the Slackware `t` series, and that you do *not* want to be prompted – simply install all the archives on the Slackware `t` series diskettes. You'll be prompted to insert each diskette in the floppy drive if you're installing from diskettes. If you're installing from a hard drive or CD-ROM, no more intervention is required by you.

Manual Slackware install. This section is for people who, for one reason or another, would like to install teTeX manually from a Linux package.

Let's assume that you've assembled the Slackware teTeX distribution on floppy diskettes labelled `t1` thru `t9.` Mount the `t1` diskette like this

```
mount /dev/fd0 /mnt
```

if your Linux configuration is a standard configuration. Actually, any mount point will do. You'll simply need to substitute the appropriate path spec in the next few steps.

The next thing you want to do is create the teTeX top-level directory. teTeX's internal paths are specified relative to its binary program files, but the Slackware distribution is archived relative to the root directory. So the top-level teTeX directory is:

```
/usr/lib/teTeX
```

For each of the `.tgz` archive files in the distribution, copy the archive file to the `/usr/lib/teTeX` directory and repeat the following commands:

You should be logged in as `root` and in the top-level directory, `/`, for these steps. I've used the `tb-xfig.tgz` archive for demonstration purposes. Of course, you'll want to substitute the name of whichever archive you're unpacking.

```
cp /mnt/tb-xfig.tgz /usr/lib/teTeX
tar -zxvf /usr/lib/teTeX/tb-xfig.tar  # v to see what's going on!
rm /usr/lib/teTeX/tb-xfig.tgz
```

Most Slackware packages that I've seen also include an install script, which the Slackware installer executes after unpacking the files. Look in the directory `/install` after you've unpacked the files. If there's a script there called `doinst.sh`, execute that, as `root`, by typing

```
sh < /install/doinst.sh
```

It may be alarming to watch all those filenames scrolling of the top of the screen as the archives are unpacked. Relax! Take a break, and freshen up your coffee (or grab another JOLT from the refrigerator, or otherwise replenish whatever you're drinking). There's only a few more steps you need to perform to install teTeX. They're covered in **Section 4.**

3.2.2 Debian GNU/Linux, V. 1.3.

Installing teTeX from Debian packages is truly trivial. `ftp` the most recent *stable* versions of the teTeX archive files from ftp.debian.org. The teTeX distribution is located in the directory

```
pub/debian/bo/binary-i386/tex
```

Retrieve the following Debian archive files via anonymous FTP (remembering to set *binary* mode for the transfers).

```
tetex-base_0.4pl6-5.deb
tetex-bin_0.4pl6-8.deb
tetex-dev_0.4pl6-8.deb
tetex-doc_0.4pl6-1.deb
tetex-extra_0.4pl6-4.deb
```

Once the files are safely transferred to your local hard disk, `su` to root, and install them using the `dpkg(1)` utility:

```
dpkg -i tetex-base_0.4pl6-5.deb
dpkg -i tetex-bin_0.4pl6-8.deb
dpkg -i tetex-dev_0.4pl6-8.deb
dpkg -i tetex-doc_0.4pl6-1.deb
dpkg -i tetex-extra_0.4pl6-4.deb
```

Installation will take some time, because the Debian archives contain the shell scripts necessary to check for old TeX installations, build the TeX and LaTeX `.fmt` files, build the path-searching database, and see to other configuration details. However, once they are finished, you should have an operational teTeX installation that needs only to be configured for the details of your local system; see section 4.

3.2.3 RedHat V. 4.2.

Presumably, you could install only selected portions of the teTeX RedHat distribution, but consistent with the philosophy of the other sections, it is assumed that you will eventually need all of the facilities provided by teTeX, and so you should install the complete distribution.

To install teTeX from RedHat Linux RPM packages, under RedHat Linux v. 4.2, `ftp`to sunsite.unc.edu and `cd` to the directory

```
pub/Linux/distributions/redhat/current/i386/RedHat/RPMS/
```

Set *binary* mode for the transfers and retrieve the following files:

```
tetex-0.4pl8-5.i386.rpm
tetex-latex-0.4pl8-5.i386.rpm
tetex-afm-0.4pl8-5.i38 6.rpm
tetex-dvilj-0.4pl8-5.i386.rpm
tetex-dvips-0.4pl8-5.i386.rpm
tetex-xdvi-0.4pl8-5.i386.rpm
tetex-texmf-src-0.4pl8-5.i386.rpm
```

This last file may not be strictly necessary. It contains the LaTeX sources, if you want to install LaTeX yourself. If you're thinking of upgrading LaTeX independently of the binaries in the future, this archive could be useful to have around:

Simply install the files above in the order given, using the `rpm -i` command, and proceed to the section, 4.

3.3 Ghostscript V. 5.03.

Ghostscript development is rapid, and the changes which are incorporated into every new version are significant. Therefore, it's worth the effort to install the version of Ghostscript that is available on its home page, http://www.cs.wisc.edu/~ghost.

At the time of this writing, the current version is 5.03. The Ghostscript archive for Linux is composed of the following files:

```
ghostscript-5.03gnu.tar.gz
ghostscript-5.03jpeg.tar.gz
ghostscript-5.03libpng.tar.gz
ghostscript-5.03zlib.tar.gz
ghostscript-fonts-std-5.03.tar.gz
ghostscript-fonts-other-5.03.tar.gz
```

What is Ghostscript, and why do you need it? Technically, Ghostscript is a Raster Image Processor. It translates PostScript code into many common, bit-mapped formats, like those understood by your printer or screen, whether or not they are equipped with PostScript. In practical terms, Ghostscript allows you to use Type 1 fonts, and mix text and graphics on any printer or video display that Ghostscript knows about.

The quality of the fonts which come with the program have improved steadily in the last several versions as well. Or maybe it's that more recent versions of Ghostscript have improved font rendering. In either case, this is of real benefit for Linux users, who may not be able to spend hundreds of dollars on commercial fonts. Because Ghostscript is able to read the font requests made by `dvips(1)`, Ghostscript's font library provides the fonts, not teTeX. But the font metrics files for Ghostscript's font library, which have the extension `.afm`, are already included in the teTeX distribution.

For information about using Ghostscript, see the file `use.txt` in the Ghostscript distribution, and the Linux Documentation Project's Printing-HOWTO. There's also a Ghostscript manual available from the Internet. See section 2.1

Or, install APSFILTER and let that run Ghostscript automatically. (See section 3.4).

A final, significant note: I would recommend that you compile Ghostscript for your own system, if possible. Combining different versions of Ghostscript and svgalib can quickly become confusing. The version of Ghostscript which is included in the Slackware `AP` set is version 2.6.2 and does not have X support compiled in. You might also have trouble finding the correct svgalib versions for it. There is supposedly a version of Ghostscript with X11 support in the Slackware `XAP` distribution series, and presumably in the other Linux distributions, though I haven't tried them. Compiling Ghostscript for your own system is far easier, it seems to me.

It's also important to remember that there are two Ghostscript releases in distribution: the commercial, Aladdin Ghostscript, and GNU Ghostscript, which lags behind Aladdin Ghostscript by several years. This is due to Ghostscript's unique licensing arrangement. See the Printing-HOWTO for more information about Ghostscript licensing.

svgalib support for GNU Ghostscript 3.33 is included in a small archive which contains a .diff file. Ghostscript 3.33 for X is also configured for JPEG support, so you should include the JPEG library sources as well. The relevant archives can be found at any GNU distribution site, like ftp://prep.ai.mit.edu/pub/gnu.

3.4 APSFILTER.

There are software packages which will simplify your life, and APSFILTER is one of them. Written by Andreas Klemm, APSFILTER works with any BSD-compatible printer daemon (which means that you have the `lpd(8)` program and an `/etc/printcap` file; see below), and provides transparent printer support for ASCII, DVI, and PostScript files, as well as files compressed by `gzip(1)`, `compress(1)`, and other data compression software.

Once you have successfully installed APSFILTER, you can print a PostScript file to whatever printer you have, by typing

```
lpr file.ps
```

Or, to print an ASCII file without PostScript translation, you can type

```
lpr -Praw file.asc
```

Amazing.

APSFILTER is surprisingly easy to install, considering that it works with many disparate elements of your system. Installing the generic APSFILTER distribution, however, does require that you have a current `gcc(1)` compiler on hand, because APSFILTER builds some of its filters during installation. Some distributions of Linux, however, provide a pre-built version, so check your specific distribution first.

In any event, you will need a correctly installed Ghostscript and `lpd(8)` installation for APSFILTER to work. The most recent APSFILTER is located in the Linux Archives at ftp://sunsite.unc.edu/pub/Linux/system/printing/.

3.5 The `lpd(8)` daemon.

There are wide variations in printers and configurations. Setting up a working printer daemon is no mean feat. If you're using teTeX on an individual system, you could simply dump the output to the printer device driver file, but this is less than desirable. You lose the filtering capabilities of the printer daemon. If you're printing on a network, having a working printer daemon is a must.

The basic UNIX program for printer management on BSD-style systems is `lpd(8)`. When you print a file with `lpr(1)` you are really sending the file to a print queue. `lpd(8)` prints files in the order they're queued. Other printer utilities include `lpq(1)`, which displays the contents of the print queue, and `lprm(1)`, which removes (dequeues) files from the print queue.

The printer daemon can perform other tasks, like transparently filtering output from various programs (using filter programs like APSFILTER, above), accept print jobs from other machines on a network, send print jobs to various printers if you have more than one connected, and hold print output until you've refilled the paper feed tray.

The Printing-HOWTO explains the process of setting up a working printer daemon in detail. Many Linux distributions already have configured `lpd(8)` suites. Check there first, because it will save you considerable work. They're usually archived, strangely enough, using the name `lpr`, so search for that program. There is also a printer daemon suite available from the Linux archives, at ftp://sunsite.unc.edu/pub/Linux/system/printing.

4 Post-installation configuration details.

The first thing you'll want to do is look at Thomas Esser's `README` file. It contains a lot of hints on how to configure teTeX for your output device (i.e., printer). The `README` file is located in the directory

```
/usr/lib/teTeX/texmf/doc/tetex
```

Read the file over with the command (the path in the following examples is that of the Slackware distribution):

```
less /usr/lib/teTeX/texmf/doc/tetex/README
```

or, print it out with the command

```
cat /usr/lib/teTeX/texmf/doc/tetex/README >/dev/lp0
```

assuming that your printer is connected to `/dev/lp0`. Substitute the device driver file that your printer is connected to, as appropriate.

Or, better still, print it using the `lpr(1)` command:

```
lpr /usr/lib/teTeX/texmf/doc/tetex/README
```

You should have installed the printer daemon that is included with your distribution of Linux. If not, do that now, per the instructions that come with the package. If you don't have one of the packages, or want to install a printer daemon yourself, see section 3.5

Print out the `teTeX-FAQ`. Keep the FAQ handy because it contains useful hints for configuring teTeX's output drivers for your printer. We'll get to that in a moment. In more recent releases of teTeX, the `teTeX-FAQ` is viewable via the `texconfig` utility.

Next, you want to define a directory to store your own TeX format files. teTeX searches the directories listed by the `$TEXINPUTS` environment variable for local TeX input files. On Chanel3, I added the line

```
export TEXINPUTS=".:~/texinputs:"
```

to the system-wide `/etc/profile` file. Individual users can set their own local `$TEXINPUTS` directory, by adding the line in their `~/.profile` or `~/.bash_profile` if `bash(1)` is the default shell. The `$TEXINPUTS` environment variable tells teTeX to look for users' individual TeX style files in the `~/texinputs` directories under each user's home directory. It is *critical* that a colon appear before and after this directory. teTeX is going to append its own directory searches to your own. You want to have teTeX search the local format files first, so it uses the local versions of any of the standard files you have edited.

Add the `/usr/lib/teTeX/bin` directory to the system-wide path if you're installing teTeX as root. Again, if you're installing a personal copy of teTeX, add the directory where the teTeX binaries are located to *the front* your `$PATH` with the following line in your `~/.profile` or `~/.bash_profile`:

```
export PATH="~/tetex/bin:"$PATH
```

Now, log in as `root` and run `texconfig` per the instructions in the `teTeX-FAQ` and choose the printer that is attached to your system. Make sure that you configure teTeX for both the correct printer and printer resolution.

Finally, run the `texhash` program. This ensures that teTeX's internal database is up to date. The database is actually a `ls-lR` file. You *must* run `texhash` every time you change the system configuration, or teTeX will not be able to locate your changes.

4.1 What if my printer isn't included?

The teTeX distribution comes with only a limited selection of DVI output drivers: `dvips(1)`, drivers for Hewlett Packard LaserJets, and nothing else. You have two options if you have a printer which isn't LaserJet-compatible: You can use `dvips(1)` and Ghostscript, which I would recommend anyway, for reasons already mentioned, or you can investigate other dviware sources.

A limited number of DVI drivers have been ported to Linux and are available as pre-built binaries. They are located in the Linux archives at ftp://sunsite.unc.edu/pub/Linux/apps/tex/dvi/.

The master dviware libraries are maintained at the University of Utah archives. If you can't find a DVI driver there that supports your printer, chances are that it doesn't exist. You can also write your *own* DVI driver using the templates available there. The library's URL is ftp://ftp.math.utah.edu/pub/tex/dvi/.

5 Using teTeX.

Theoretically, at least, everything is installed correctly and is ready to run. teTeX is a very large software package. As with any complex software package, you'll want to start by learning teTeX slowly, instead of being overwhelmed by its complexity.

At the same time, we want the software to do something useful. So instead of watching TeX typeset

```
``Hello, World!''
```

as Professor Knuth suggests in the *The TeXBook,* we'll produce a couple of teTeX's own documents in order to test it.

The next section, 5.1, is really a tutorial for operating teTeX. It covers printing the documentation included with teTeX (which is in LaTeX and `.dvi` format, of course). The following section, 5.2, is more of a "cookbook" than a tutorial. It discusses how to format LaTeX documents, and covers a few of the commands and environments of the more commonly used document classes.

The section 5.3 tells how to use the many pre-existing LaTeX packages to customize documents to your specifications.

5.1 Printing the documentation.

You should be logged in as `root` the first few times you run teTeX. If you aren't, metafont may not be able to create the necessary directories for its fonts. The `texconfig` program includes an option to make the font directories world-writable, but if you're working on a multi-user system, security considerations may make this option impractical or undesirable.

In either instance, if you don't have the appropriate permissions to write to the directories where the fonts are stored, `metafont` will complain loudly because it can't make the directories. You won't see any output because you have a bunch of zero-length font characters. This is no problem. Simply log out, re-login as `root,` and repeat the offending operation.

The nice thing about teTeX is that, if you blow it, no real harm is done. It's not like a compiler, where, say, you will trash the root partition if a pointer goes astray. What, you haven't read the teTeX manual yet? Of course you haven't. It's still in the distribution, in source code form, waiting to be output.

So, without further delay, you will want to read the teTeX manual. It's located in the directory

```
/usr/lib/teTeX/texmf/doc/tetex.
```

The LaTeX source for the manual is called `TETEXDOC.tex`. (The `.tex` extension is used for both TeX and LaTeX files. Some editors, like `emacs(1),` can tell the difference.) There is also a file `TETEXDOC.dvi` included with the distribution, which you might want to keep in a safe place—say, another directory —in case you want to test your `.dvi` drivers later. With that out of the way, type

```
latex TETEXDOC.tex
```

LaTeX will print several warnings. The first,

```
LaTeX Warning: Label(s) may have changed. Rerun to get the
cross-references right.
```

is standard. It's common to build a document's Table of Contents by LaTeXing the document twice. So, repeat the command. The other warnings can be safely ignored. They simply are informing you that some of the FTP paths mentioned in the documentation are too wide for their alloted spaces. (If you're really inquisitive, look at one of the TeX references for a discussion of `\hbox` and `\vbox.`)

teTeX will have generated several files from `TETEXDOC.tex.` The one that we're interested in is `TETEXDOC.dvi.` This is the device-independent output which you can send either to the screen or the printer. If you're running teTeX under the X Windows System, you can preview the document with `xdvi(1).`

For the present, let's assume that you have a HP LaserJet II. You would give the command

```
dvilj2 TETEXDOC.dvi
```

which will write a PCL output file from `TETEXDOC.dvi,` including soft fonts which will be downloaded to the LaserJet. This is *not* a feature of TeX or LaTeX, but a feature provided by `dvilj2(1).` Other `.dvi` drivers provide features which are relevant to the devices they support. `dvilj2(1)` will fill the font requests which were made in the original LaTeX document with the the closest equivalents available on the system. In the case of a plain-text document like `TETEXDOC.tex,` there isn't much difficulty. All of the fonts requested by `TETEXDOC.tex` will be generated by metafont, which is automatically invoked by `dvilj2(1)` and generates the fonts if they aren't already present. (If you're running `dvilj2(1)` for the first time, the program needs to generate all of the fonts, which could take up to several days if you're using a *really* slow machine.) There are several options which control font generation via `dvilj2(1)`; they're outlined in the manual page. At this point, you shouldn't need to operate metafont directly. If you do, then something has gone awry with your installation. All of the `.dvi` drivers will invoke metafont directly via the kpathsea path-searching library—also beyond the scope of this document—and you don't need to do any more work with metafont for the present—all of the metafont sources for the Computer Modern font library are provided.

You can print `TETEXDOC.lj` with the command

```
lpr TETEXDOC.lj
```

You may need to install a printer filter that understands PCL. Look at the Printing-HOWTO for details.

The nine-page *teTeX Guide* provides some useful information for further configuring your system, some of which I have mentioned, much which this document doesn't cover.

Some of the information in the next section I haven't been able to test, because I have a non-PostScript HP Deskjet 400 color ink jet printer connected to Chanel3's parallel port. However, not owning a PostScript printer is no barrier to printing text and graphics from your text documents. See the section 3.3 to install Ghostscript, if it isn't already installed on your system.

5.2 TeX and LaTeX commands.

5.2.1 Document structure.

Preparing documents for TeX typesetting is easy. Make sure there's a blank line between the paragraphs of a plain text file, and run file through the TeX program with the command

```
TeX your_text_file
```

The result will be a file of the same base name and the extension `.dvi`. The text is set in 10-point, Computer Modern Roman, single-spaced, with justified left and right margins. If you receive error messages from special characters like dollar signs, escape them with a backslash character, \, and run TeX on the file again. You should be able to process the resulting file with the `.dvi` file translator of your choice (see above) to get printed output.

The only other peculiarity of TeX input files is to make sure that you use opening and closing quotes, which are denoted in the input file with the grave accent and single quote characters. Emacs' TeX mode will do this for you automatically.

```
"These are ascii-type quotes."
``These are `TeX-style' quotes.''
```

You can consult a guide like *A Gentle Introduction to TeX,* described above, for hints on how to make modifications to the default TeX page format.

Documents formatted for LaTeX have a few more rules, but with complex documents, LaTeX can greatly simplify the formatting process.

Essentially, LaTeX is a document markup language which tries to separate the output style from the document's logical content. For example, formatting a section heading with TeX would require specifying 36 points of white space above the heading, then the heading itself set in bold, 24-point type, then copying the heading text and page number to the Table of Contents, then leaving 24 points of white space after the heading. By contrast, LaTeX has the `\section{}` command, which does all of the work for you. If you need to change the format of the section headings throughout your document, you can change the definition of `\section{}` instead of the text in the document. You can see where this would save hours of reformatting for documents of more than a dozen pages in length.

All LaTeX documents have three sections: a *preamble,* the *body* text, and a *postamble*. These terms are standard jargon and are widely used by TeXperts.

The preamble, at a minimum, specifies the type of document to be produced—the *document class*—and a statement which signals the beginning of the document's body text. For example:

```
\documentclass{article}
\begin{document}
```

The document's postamble is usually very simple. Except in specialized cases, it contains only the statement:

```
\end{document}
```

Note the `\begin{document}` and `\end{document}` pairing. In LaTeX, this is called an *environment*. All text must appear within an environment, and many commands are effective only in the environments in which they're called. The `document` environment is the only instance where LaTeX enforces this convention, however. That is, it's the only environment that is required in a document. (An exception is `letter` class, which also requires you to declare `\begin{letter}` and `\end{letter}`. See the section 5.2.4.) However, many formatting features are specified as environments. They're described in the following sections.

The document classes can be called with arguments. For example, instead of the default, 10-point type used as the base point size, as in the previous example, we could have specified

```
\documentclass[12pt]{article}
```

to produce the document using 12 points as the base point size. The document class, *article,* makes the necessary adjustments.

There are a few document classes which are commonly used. They're described below. The *report* class is similar to *article* class, but produces a title page and starts each section on a new page. The *letter* class includes special definitions for addresses, salutations, and closings, a few of which are described below.

You can include canned LaTeX code, commonly known as a *package,* with the `\usepackage{}` command.

```
\usepackage{fancyhdr}
```

The command above would include the LaTeX style file `fancyhdr.sty` from one of the `TEXINPUTS` directories, which you and teTeX specified during installation and setup processes.

```
\documentclass{article}
\usepackage{fancyhdr}
\begin{document}
```

Note that the `\usepackage{}` declarations are given before the `\begin{document}` statement; that is, in the document preamble.

`fancyhdr.sty` extends the `\pagestyle{}` command so that you can create custom headers and footers. Most LaTeX document classes provide headers and footers of the following standard page styles:

```
\pagestyle{plain}       % default pages style -- page number centered at
                        % the bottom of the page.
\pagestyle{empty}       % no headers or footers
\pagestyle{headings}    % print section number and page number at the
                        % top of the page.
\pagestyle{myheadings}  % print custom information in the page heading.
```

Everything on a line to the right of the percent sign is a comment.

The `\pagestyle{}` command doesn't take effect until the following page. To change the headers and footers on the current page, use the command

```
\thispagestyle{the_pagestyle}
```

5.2.2 Characters and type styles.

Character styles are partially a function of the fonts specified in the document. However, bold and italic character emphasis should be available for every font present on the system. Underlining, too, can be used, though its formatting presents special problems. See section 5.3, below.

You can specify text to be emphasized in several ways. The most portable is the `\em` command. All text within its scope is italicized by default. For example:

```
This word will be {\em emphasized.}
```

If you have italicized text that runs into text which is not italicized, you can specify an italic correction factor to be used. The command for this is `\/`; that is, a backslash and a forward slash.

```
This example {\em will\/} print correctly.

This example will {\em not} print correctly.
```

Slightly less portable, but still acceptable in situations where they're used singly, are the commands `\it`, `\bf`, and `\tt`, which specify that the characters within their scope be printed using italic, bold, and monospaced (teletype) typefaces, respectively.

```
{\tt This text will be printed monospaced,}
{\it this text will be italic,} and
{\bf this text will be bold\dots} all in one paragraph.
```

The command `\dots` prints a series of three periods for ellipses, which will not break across a line.

The most recent version of LaTeX, which is what you have, includes commands which account for instances where one emphasis command would supersede another.

```
This is {\it not {\bf bold italic!}}
```

What happens is that teTeX formats the text with the italic typeface until it encounters the `\bf` command, at which point it switches to boldface type.

To get around this, the NFSS scheme of selecting font shapes requires three parameters for each typeface: shape, series, and family. Not all font sets will include all of these styles. LaTeX will print a warning, however, if it needs to substitute another font.

You can specify the following font shapes:

```
\textup{text}          % upright shape (the default)
\textit{text}          % italic
\textsl{text}          % slanted
\textsc{text}          % small caps
```

These are the two series that most fonts have:

```
\textmd{text}          % medium series (the default)
\textbf{text}          % boldface series.
```

There are generally three families of type available.

```
\textrm{text}          % roman (the default)
\textsf{text}          % sans serif
\texttt{text}          % typewriter (monospaced, Courier-like)
```

Setting font styles using these parameters, you can combine effects.

```
\texttt{\textit{This example likely will result in a font
substitution, because many fonts don't include a typewriter italic
typeface.}}
```

The font family defaults to Computer Modern, which is a bit-mapped font. Other font families are usually PostScript-format Type 1 fonts. See section 7 for details on how to specify them.

There are also many forms of accents and special characters which are available for typesetting. This is only a few of them. (Try typesetting these on your own printer.)

```
\'{o}   \`{e}   \^{o}   \"{u}   \={o}   \c{c}   `? `!
\copyright      \pounds         \dag
```

Finally, there are characters which are used as meta- or escape characters in TeX and LaTeX. One of them, the dollar sign, is mentioned above. The complete set of metacharacters, which need to be escaped with a backslash to be used literally, is:

```
# $ % & _ { }
```

There are also different alphabets available, like Greek and Cyrillic. LaTeX provides many facilities for setting non-English text, which are covered by some of the other references mentioned here

5.2.3 Margins and line spacing.

Changing margins in a TeX or LaTeX document is not a straightforward task. A lot depends on the relative indent of the text you're trying to adjust the margin for. The placement of the margin-changing command is also significant.

For document-wide changes to LaTeX documents, the `\evensidemargin` and `\oddsidemargin` commands are available. They affect the left-hand margins of the even-numbered and odd-numbered pages, respectively. For example,

```
\evensidemargin=1in
\oddsidemargin=1in
```

adds on inch to the left-hand margin of the even and odd pages *in addition* to the standard one-inch, left-hand margin. These commands affect the entire document and will shift the entire body of the text right and left across a page, regardless of any local indent, so they're safe to use with LaTeX environments like `verse` and `list`.

Below is a set of margin-changing macros which I wrote. They have a different effect than the commands mentioned above. Because they use plain TeX commands, they're not guaranteed to honor the margins of any LaTeX environments which may be in effect, but you can place them anywhere in a document and change the margins from that point on.

```
%%  margins.sty -- v. 0.1   by Robert Kiesling
%%  Copies of this code may be freely distributed in verbatim form.
%%
%%  Some elementary plain TeX margin-changing commands. Lengths are
%%  in inches:
%%  \leftmargin{1}   %% sets the document's left margin in 1 inch.
%%  \leftindent{1}   %% sets the following paragraphs' indent in
%%                     1 inch.
%%  \rightindent{1}  %% sets the following paragraphs' right margins
%%                   %% in 1 inch.
%%  \llength{3}      %% sets the following lines' lengths to 3 inches.
```

```
%%
\message{Margins macros...}
\def\lmargin#1{\hoffset = #1 in}
\def\lindent#1{\leftskip = #1 in}
\def\rindent#1{\rightskip = #1 in}
\def\llength#1{\hsize = #1 in}
%%
%% (End of margins macros.}
```

Place this code in a file called `margins.sty` in your local `$TEXINPUTS` directory. The commands are explained in the commented section of the file. To include them in a document, use the command

```
\usepackage{margins}
```

in the document preamble.

While we're on the subject, if you don't want the right margin to be justified, which is the default, you can tell LaTeX to use ragged right margins by giving the command:

```
\raggedright
```

Setting line spacing also has its complexities.

The *baselineskip* measurement is the distance between lines of text. It is given as an absolute measurement. For example,

```
\baselineskip=24pt
```

or even better:

```
\setlength{\baselineskip}{24pt}
```

The difference between the two forms is that *setlength* will respect any scoping rules that may be in effect when you use the command.

The problem with using baselineskip is that it also affects the distance between section headings, footnotes, and the like. You need to take care that baselineskip is correct for whatever text elements you're formatting. There are, however, LaTeX macro packages, like `setspace.sty`, which will help you in these circumstances. See section 5.3.

5.2.4 Document classes.

LaTeX provides document classes which provide standardized formats for documents. They provide environments to format lists, quotations, footnotes, and other text elements. Commonly used document classes are covered in the following sections.

Articles and reports. As mentioned above, the `article` class and the `report` class are similar. The main differences are that the report class creates a title page by default and begins each section on a new page. Mostly, though, the two document classes are similar.

To create titles, abstracts, and bylines in these document classes, you can type, for example,

```
\title{The Breeding Habits of Cacti}
\author{John Q. Public}
\abstract{Description of how common desert cacti search
for appropriate watering holes to perform their breeding
rituals.}
```

in the document preamble. Then, the command

```
\maketitle
```

given at the start of the text, will generate either a title page in the report class, or the title and abstract at the top of the first page, in the article class.

Sections can be defined with commands that include the following:

```
\section
\subsection
\subsubsection
```

These commands will produce the standard, numbered sections used in technical documents. For unnumbered sections, use

```
\section*
\subsection*
\subsubsection*
```

and so on.

LaTeX provides many environments for formatting displayed material. You can include quoted text with the `quotation` environment.

```
\begin{quotation}
Start of paragraph to be quoted...

... end of paragraph.
\end{quotation}
```

For shorter quotes, you can use the `quote` environment.

To format verse, use the `verse` environment.

```
\begin{verse}
Because I could not stop for death\\
He kindly stopped for me
\end{verse}
```

Notice that you must use the double backslashes to break lines in the correct places. Otherwise, LaTeX fills the lines in a verse environment, just like any other environment.

Lists come in several flavors. To format a bulleted list, the `list` environment is used:

```
\begin{list}
\item
This is the first item of the list.
\item
This is the second item of the list...
\item
... and so on.
\end{list}
```

A numbered list uses the `enumerate` environment:

```
\begin{enumerate}
\item
Item No. 1.
\item
Item No. 2.
\item
\dots
\end{enumerate}
```

A descriptive list uses the `description` environment.

```
\begin{description}
\item{Oven} Dirty, needs new burner.
\item{Refrigerator}  Dirty.  Sorry.
\item{Sink and drainboard}  Stained, drippy, cold water faucet.
\end{description}
```

Letters. The `letter` class uses special definitions to format business letters.

The `letter` environment takes one argument, the address of the letter's addressee. The `address` command, which must appear in the document preamble, defines the return address. The `signature` command defines the sender's name as it appears after the closing.

The LaTeX source of a simple business letter might look like this.

```
\documentclass[12pt]{letter}
\signature{John Q. Public}
\address{123 Main St.\\Los Angeles, CA.  96005\\Tel: 123/456-7890}
\begin{document}
\begin{letter}{ACME Brick Co.\\100 Ash St.\\San Diego, CA 96403}
\opening{Dear Sir/Madam:}

With regard to one of your bricks that I found on my living room
carpet surrounded by shards of my broken front window...

(Remainder of the body of the letter.)

\closing{Sincerely,}

\end{letter}
\end{document}
```

Note that the addresses include double backslashes, which specify where the line breaks should occur.

5.3 LaTeX extension packages and other resources.

We mentioned above that using underlining as a form of text emphasis presents special problems. Actually, TeX has no problem underlining text, because it is a convention of mathematical typesetting. In LaTeX, you can underline words with the command:

```
\underline{text to be underlined}
```

The problem is that underlining will not break across lines, and, in some circumstances, underlining can be uneven. However, there is a LaTeX macro packagem, ready-made, that makes underlining the default mode of text emphasis. It's called `ulem.sty`, and is one of the many contributed LaTeX packages that are freely available via the Internet.

To use `ulem.sty`, include the command

```
\usepackage{ulem}
```

in the document preamble.

The LaTeX Catalogue provides one-line descriptions of every LaTeX package available, their names and CTAN paths. For the URL of the most current edition of the Catalogue, see the section 2.1.

The packages which are available for LaTeX include:

ifthen

Include conditional statements in your documents.

initials

Defines a font for initial dropped capitals.

sanskrit

Font and preprocessor for producing documents in Sanskrit.

recipe

A LaTeX2e class to typeset recipes.

refman

Variant report and article styles.

To make the path given in the Catalogue into a fully-qualified URL, concatenate the path to the hostname URL and top-level path of the CTAN archive you wish to contact. For example, the top-level CTAN directory of the site ftp.tex.ac.uk is `ctan/tex-archive`. The complete URL of the directory of the **refman** package would be:

```
ftp://ftp.tex.ac.uk/ctan/tex-archive/   +
macros/latex/contrib/supported/refman   =

ftp://ftp.tex.ac.uk/ctan/tex-archive/macros/latex/contrib/supported/refman/
```

Some packages have more than one file, so only the path to the package's directory is given.

When you have the URL in hand, you can retrieve the package from one of the CTAN archive sites listed in section 8. You can download a complete list of the archive's contents as the file `FILES.byname`, in the archive's top-level directory. You can also search the archive on line for a keyword with the `ftp(1)` command

```
quote site index <keyword>
```

6 Mixing text and graphics with `dvips(1)`.

In general, this section applies to any TeX or LaTeX document which mixes text and graphics. teTeX, like most other TeX distributions, is configured to request Computer Modern fonts by default. When printing documents with Type 1 scalable fonts or graphics, font and graphics imaging is the job of `dvips(1)`. `dvips(1)` can use either Computer Modern bit mapped fonts or Type 1 scalable fonts, or any combination of the two. First, let's concentrate on printing and previewing some graphics.

In general, you will want to follow this procedure any time a LaTeX source document has the statement

```
\includepackage{graphics}
```

in the document preamble. This statement tells LaTeX to include the text of the `graphics.sty` package in the source document. There are other commands to perform graphics operations, and the statements in plain-TeX documents may not clue you in whether you need to use `dvips(1).` The difference will be apparent in the output, though, when the document is printed with missing figures and other graphics.

So, for now, we'll concentrate on printing documents which use the LaTeX `graphics.sty` package. You might want to take a look at the original TeX input. It isn't included in the teTeX distribution, but it is available at

```
~CTAN/macros/latex/packages/graphics/grfguide.tex.
```

What the teTeX distribution does include is the `.dvi` output file, and it is already TeXed for you. There is a reason for this, and it has to do with the necessity of including Type 1 fonts in the output in order for the document to print properly. If you want to LaTeX `grfguide.tex,` see the next section. For now, however, we'll work on getting usable output using `dvips(1).`

The file `grfguide.dvi` is located in the directory

```
texmf/doc/latex/graphics
```

The first step in outputting `grfguide.dvi` is to translate it to PostScript. The program `dvips(1)` is used for this. It does just exactly what its name implies. There are many options available for invoking `dvips(1),` but the simplest (nearly) form is

```
dvips -f -r <grfguide.dvi >grfguide.ps
```

The `-f` command switch tells `dvips(1)` to operate as a filter, reading from standard input and writing to standard output. `dvips(1)` output can be configured so its output defaults to `lpr(1).` (Mine does, which allows me to print directly from `dvips(1)`.) Post processors like Ghostscript and printing filters like APSFILTER (see section 3.4 name="APSFILTER"), can be configured for your own needs. If you need to feed the output manually to a post-processor, the `-f` option is generally the first you should include in the `dvips(1)` command line. This form also seems to be easier to use in shell scripts.

If you can print PostScript directly to your printer via `lpr(1),` you can simply type

```
dvips -r grfguide.dvi
```

The `-r` option tells dvips to output the pages in reverse order so they stack correctly when they exit a printer. Use it or not, as appropriate for your output device.

Depending on whether you still have the fonts that `dvilj2(1)` generated from the last document, `dvips(1)` and metafont may or may not need to create new fonts needed by `grfguide.dvi.` Eventually, though, `dvips(1)` will output a list of the pages translated to PostScript, and you will have your PostScript output ready to be rendered on whatever output device you have available.

If you're lucky (and rich), then you have a PostScript-capable printer already and will be able to print `grfguide.ps` directly. You can either spool the output to the printer using `lpr(1)`. If for some reason your printer software doesn't work right with PostScript files, you can, in a pinch, simply dump the file to printer, with

```
cat grfguide.ps >/dev/lp0
```

or whichever port your printer is attached to, though this is not recommended for everyday use.

If you want or need to invoke Ghostscript manually, this is the standard procedure for its operation. The first thing you want to do is invoke Ghostscript to view its command line arguments, like this:

```
gs -help | less
```

You'll see a list of supported output devices and sundry other commands. Pick the output device which most nearly matches your printer. On Chanel3, because I generally produce black-and-white text, I use the `cdjmono` driver, which drives a color Deskjet in monochrome (black and white) mode.

The command line I would use is:

```
gs -dNOPAUSE -sDEVICE=cdjmono -sOutputFile=/tmp/gs.out grfguide.ps -c quit
```

This will produce my HP-compatible output in the `/tmp` directory. It's a good idea to use a directory like `/tmp`, because `gs(1)` can be particular about access permissions, and you can't (and shouldn't) always count on being logged in as `root` to perform these steps. Now you can print the file:

```
lpr /tmp/gs.out
```

Obviously, this can all go into a shell script. On my system, I have two simple scripts written, `pv` and `pr`, which simply outputs the PostScript file either to the display or the printer. Screen previewing is possible without X, but it's far from ideal. So, it's definitely worth the effort to install XFree86, or TinyX (which is what I did) to view the output on the screen.

The order of commands in a `gs(1)` command line is significant, because some of the options tell Ghostscript to look for pieces of PostScript code from its library.

The important thing to remember is that `grfguide.dvi` makes requests for both Computer Modern bit mapped and Type 1 scaled fonts. If you can mix scalable and bit mapped fonts in a document, you're well on the way to becoming a TeXpert.

7 Using PostScript fonts.

It used to be that public domain, Type 1 fonts were much poorer quality than Computer Modern bit mapped fonts. This situation has improved in the last several years, though, but matching the fonts is up to you. Having several different font systems on one machine can seem redundant and an unnecessary waste of disk space. And the Computer Modern fonts can seem, well, a little too *formal* to be suitable for everyday use. It reminds me sometimes of bringing out the good China to feed the dog. At least you don't need to spend a bundle on professional quality fonts any longer.

One of the major improvements of LaTeX2e over its predecessor was the inclusion of the New Font Selection Scheme. (It's now called PSNFSS.) Formerly, TeX authors would specify fonts with commands like

```
\font=bodyroman = cmr10 scaled \magstep 1
```

which provides precision but requires the skills of a type designer and mathematician to make good use of. Also, it's not very portable. If another system didn't have the font `cmr10` (this is TeX nomenclature for Computer Modern Roman, 10 point, with the default medium stroke weight), somebody would have to re-code the fonts specifications for the entire document. PSNFSS, however, allows you specify fonts by family (Computer Modern, URW Nimbus, Helvetica, Utopia, and so forth), weight (light, medium, bold), orientation (upright or oblique), face (Roman, Italic), and base point size. (See the section 5.2.2 for a description of the commands to specify typefaces.) Many fonts are packaged as families. For example, a Roman-type font may come packaged with a sans serif font, like Helvetica, and a monospaced font, like Courier. You, as the author of a LaTeX document, can specify an entire font family with one command.

There are, as I said, several high-quality font sets available in the public domain. One of them is Adobe Utopia. Another is Bitstream Charter. Both are commercial quality fonts which have been donated to the public domain.

These happen to be two of my favorites. If you look around one of the CTAN sites, you will find these and other fonts archived there. There are enough fonts around that you'll be able to design documents the way you want them to look, and not just English text, either. TeX was originally designed for mathematical typesetting, so there is a full range of mathematical fonts available, as well as Cyrillic, Greek, Kana, and other alphabets too numerous to mention.

The important thing to look for is files which have either the `.pfa` or `.pfb` extension. They indicate that these are the scalable fonts themselves, not simply the metrics files. Type 1 fonts use `.pfm` metric files, as opposed to the `.tfm` metric files which bit mapped fonts use. The two font sets I mentioned above are included in teTeX distributions, as well as separately.

What I said above, concerning the ease of font selection under PSNFSS, is true in this instance. If we want to use the Charter fonts in our document instead of Computer Modern bit mapped, all that is necessary is include the LaTeX statement

```
\renewcommand{\familydefault}{bch}
```

in the document preamble, where "bch" is the common designation for Bitstream Charter. The Charter fonts reside in the directory

```
/usr/lib/teTeX/texmf/fonts/type1/bitstrea/charter
```

There you'll see the `.pfb` files of the Charter fonts: `bchb8a.pfb` for Charter Bold, `bchr8a.pfb` for Charter Roman, `bchbi8a.pfb` for Charter Bold Italic. The "8a" in the font names indicates the character encoding. At this point you shouldn't need to worry much about them, because the encodings mostly differ for 8-bit characters, which have numeric values above 128 decimal. They mostly define accents and non-English characters. The Type 1 font encodings generally work well for Western alphabets because they conform to the ISO 8859 standards for international character sets, so this is an added benefit of using them.

To typeset a document which has Charter fonts selected, you would give the command

```
pslatex document.tex
```

`pslatex` is a variant of teTeX's standard `latex(1)` command which defines the directories where the Type 1 fonts are, as well as some additional LaTeX code to load. You'll see the notice screen for `pslatex` followed by the status output of the TeX job itself. In a moment, you'll have a `.dvi` file which includes the Charter font requests. You can then print the file with `dvips(1)`, and `gs(1)` if necessary.

Installing a Type 1 font set is not difficult, as long as you follow a few basic steps. You should unpack the fonts in a subdirectory of the `/usr/lib/teTeX/texmf/fonts/type1` directory, where your other Type 1 fonts are located, and then run `texhash` to let the directory search routines know that the fonts have been added. Then you need to add the font descriptions to the file `psfonts.map` so `dvips(1)` knows they're on the system. The format of the `psfonts.map` file is covered in a couple different places in the references mentioned above. Again, remember to run the `texhash` program to update the teTeX directory database.

It is definitely an advantage to use the X Windows System with teTeX– XFree86 under Linux – because it allows for superior document previewing. It's not required, but in general, anything that allows for easier screen previewing is going to benefit your work, in terms of the quality of the output. However, there is a tradeoff with speed of editing, which is much quicker on character-mode displays. Having an editor which is slower than molasses in Minnesota can definitely hinder your work.

Anyway, whether or not you are able to view documents easily on-screen, please recycle your paper, and use both sides of each sheet. If possible, purchase recycled photocopy paper to print on. You don't want your workplace to look like a branch office of a paper company.

Remember: Save a tree... kill an editor.

Robert Kiesling

kiesling@terracom.net

8 Appendix: CTAN Site Listing

This is the text of the file `CTAN.sites`, which is available in the top-level directory of each CTAN archive or mirror site.

```
In order to reduce network load, it is recommended that you use the
Comprehensive TeX Archive Network (CTAN) host which is located in the
closest network proximity to your site.  Alternatively, you may wish to
obtain a copy of the CTAN via CD-ROM (see help/CTAN.cdrom for details).

Known mirrors of the CTAN reside on (alphabetically):
  cis.utovrm.it (Italia)               /TeX
  ctan.unsw.edu.au (NSW, Australia)    /tex-archive
  dongpo.math.ncu.edu.tw (Taiwan)      /tex-archive
  ftp.belnet.be (Belgium)              /packages/TeX
  ftp.ccu.edu.tw (Taiwan)              /pub/tex
  ftp.cdrom.com (West coast, USA)      /pub/tex/ctan
  ftp.comp.hkbu.edu.hk (Hong Kong)     /pub/TeX/CTAN
  ftp.cs.rmit.edu.au  (Australia)      /tex-archive
```

```
  ftp.cs.ruu.nl (The Netherlands)       /pub/tex-archive
  ftp.cstug.cz (The Czech Republic)     /pub/tex/CTAN
  ftp.duke.edu (North Carolina, USA)    /tex-archive
  ftp.funet.fi (Finland)                /pub/TeX/CTAN
  ftp.gwdg.de (Deutschland)             /pub/dante
  ftp.jussieu.fr (France)               /pub4/TeX/CTAN
  ftp.kreonet.re.kr (Korea)             /pub/CTAN
  ftp.loria.fr (France)                 /pub/unix/tex/ctan
  ftp.mpi-sb.mpg.de (Deutschland)       /pub/tex/mirror/ftp.dante.de
  ftp.nada.kth.se (Sweden)              /pub/tex/ctan-mirror
  ftp.oleane.net (France)               /pub/mirrors/CTAN/
  ftp.rediris.es (Espa\~na)             /mirror/tex-archive
  ftp.rge.com (New York, USA)           /pub/tex
  ftp.riken.go.jp (Japan)               /pub/tex-archive
  ftp.tu-chemnitz.de (Deutschland)      /pub/tex
  ftp.u-aizu.ac.jp (Japan)              /pub/tex/CTAN
  ftp.uni-augsburg.de (Deutschland)     /tex-archive
  ftp.uni-bielefeld.de (Deutschland)    /pub/tex
  ftp.unina.it (Italia)                 /pub/TeX
  ftp.uni-stuttgart.de (Deutschland)    /tex-archive (/pub/tex)
  ftp.univie.ac.at (\"Osterreich)       /packages/tex
  ftp.ut.ee (Estonia)                   /tex-archive
  ftpserver.nus.sg (Singapore)          /pub/zi/TeX
  src.doc.ic.ac.uk (England)            /packages/tex/uk-tex
  sunsite.auc.dk (Denmark)              /pub/tex/ctan
  sunsite.cnlab-switch.ch (Switzerland) /mirror/tex
  sunsite.icm.edu.pl (Poland)           /pub/CTAN
  sunsite.unc.edu (North Carolina, USA) /pub/packages/TeX
  wuarchive.wustl.edu (Missouri, USA)   /packages/TeX

Known partial mirrors of the CTAN reside on (alphabetically):
  ftp.adfa.oz.au (Australia)            /pub/tex/ctan
  ftp.fcu.edu.tw (Taiwan)               /pub2/tex
  ftp.germany.eu.net (Deutschland)      /pub/packages/TeX
  ftp.gust.org.pl (Poland)              /pub/TeX
  ftp.jaist.ac.jp (Japan)               /pub/TeX/tex-archive
  ftp.uu.net (Virginia, USA)            /pub/text-processing/TeX
  nic.switch.ch (Switzerland)           /mirror/tex
  sunsite.dsi.unimi.it (Italia)         /pub/TeX
  sunsite.snu.ac.kr (Korea)             /shortcut/CTAN

Please send updates to this list to <ctan@urz.uni-heidelberg.de>.

The participating hosts in the Comprehensive TeX Archive Network are:
```

```
ftp.dante.de  (Deutschland)
     -- anonymous ftp                 /tex-archive (/pub/tex /pub/archive)
     -- gopher on node gopher.dante.de
     -- e-mail via ftpmail@dante.de
     -- World Wide Web access on www.dante.de
     -- Administrator: <ftpmaint@dante.de>

ftp.tex.ac.uk (England)
     -- anonymous ftp                 /tex-archive (/pub/tex /pub/archive)
     -- gopher on node gopher.tex.ac.uk
     -- NFS mountable from nfs.tex.ac.uk:/public/ctan/tex-archive
     -- World Wide Web access on www.tex.ac.uk
     -- Administrator: <ctan-uk@tex.ac.uk>
```