5 CTAN, packages, and online help

The Comprehensive TeX Archive Network (CTAN) is a repository of Web documents and files from HyperText Transfer Protocol (HTTP) and File Transfer Protocol (FTP) servers worldwide which contain copies of almost every piece of free software related to TeX and LaTeX.

CTAN is based on three main servers, and there are several online indexes available. There are complete TeX and LaTeX systems for all platforms, utilities for text and graphics processing, conversion programs into and out of LaTeX, printer drivers, extra typefaces, and (possibly the most important) the LaTeX packages. The three main servers are:

□ TeX Users Group: `http://www.ctan.org/`

□ UK TeX Users Group: `http://www.tex.ac.uk/`

Always try CTAN first

CTAN should always be your first port of call when looking for a software update or a feature you want to use. Please don't ask the network help resources (➠ § 5.3) until you have checked CTAN and the FAQ (§ 5.3.1).

□ Deutschsprachige Anwendervereinigung TEX e.V. (DANTE, the German-speaking TEX Users Group); `http://dante.ctan.org/`

5.1 Packages

Add-on features for LATEX are known as *packages*. Dozens of these are pre-installed with LATEX and can be used in your documents immediately. They are all be stored in subdirectories of `texmf/tex/latex` (or its equivalent) named after each package. To find out what other packages are available and what they do, you should use the CTAN search page[1] which includes a link to Graham Williams' comprehensive package catalogue.

A package is a file or collection of files containing extra LATEX commands and programming which add new styling features or modify those already existing. Installed package files all end with `.sty` (they used to be called 'style files') and there may be ancillary files as well.

When you try to typeset a document which requires a package which is not installed on your system, LATEX will warn you with an error message that it is missing (see § 4.2.3.7), and you can then download the package and install it using the instructions in § 5.2.

Auto-install of missing packages MIKTEX has a feature that spots if you try to use a package that isn't installed, and offers to download and install it for you there and then. This is now being ported to other distributions of LATEX, so check the documentation on the DVD to see if it is working in your installation. This avoids you having to do manual installation except for a few packages that are very old or do not conform to the TDS standard.

You can also download updates to packages you already have, both the ones that were installed along with your version of LATEX as well as ones you have added.

There is no limit to the number of packages you can have installed on your computer (apart from disk space!), but there is probably a physical limit to the number that can be used inside any one LATEX document at the same time, although it depends on how big each package is. In practice there is no problem in having even a couple of dozen packages active (the style file for this document uses over 30).

5.1.1 Using an existing package

To use a package already installed on your system, insert a **`\usepackage`** command in your document preamble with the package name in curly braces, as we have already seen in earlier chapters. For example, to use the xcolor package, which lets you typeset in colours (I warned you this was coming!), you would type:

[1] `http://www.ctan.org/search`

```
\documentclass[11pt,a4paper,oneside]{report}
\usepackage{xcolor}
\begin{document}
...
\end{document}
```

You can include several package names in one `\usepackage` command by separating the names with commas, and you can have more than one `\usepackage` command.

Some packages allow optional settings in square brackets. If you use these, you must give the package its own separate `\usepackage` command, like geometry and xcolor shown below:

```
\documentclass[11pt,a4paper,oneside]{report}
\usepackage{pslatex,palatino,avant,graphicx}
\usepackage[margin=2cm]{geometry}
\usepackage[svgnames]{xcolor}
\begin{document}

\title{\color{Crimson}Practical Typesetting}
\author{\color{DarkSlateBlue}Peter Flynn\\Silmaril Consultants}
\date{\color{ForestGreen}January 2011}
\maketitle

\end{document}
```

(Incidentally, this is a very crude and cumbersome way to do colours in titling. It's fine for a one-time document, but it will interfere with running heads if you use them; and if it's for a repeatable style we'll see in Chapter 9 how it can be automated as part of the `\maketitle` command and kept out of the author's way.)

Many packages can have additional formatting specifications in optional arguments in square brackets, in the same way as geometry and xcolor do. Read the documentation for the package concerned to find out what can be done.

5.1.2 Package documentation

To find out what commands a package provides (and thus how to use it), you need to read the documentation. The simplest way is to use your command window and type `texdoc` followed by the package name. This will bring up the documentation in your PDF or DVI viewer.

Exercise 5.1: Add colour

Use the xcolor package to add some colour to your document. Stick with primary colours for the moment.

Use the geometry package to change the margins.

Reprocess and print your document if you have a colour printer (monochrome printers should print it in shades of grey).

If that doesn't find it, in the `texmf/doc` subdirectory of your installation there should be directories full of `.dvi` and `.pdf` files, one for every package installed. These can be previewed or printed like any other DVI or PDF file (see § 4.3.1). If your installation procedure has not installed the documentation, the files can all be downloaded from CTAN.

Before using a package, you should read the documentation carefully, especially the subsection usually called 'User Interface', which describes the commands the package makes available. You cannot just guess and hope it will work: you have to read it and find out.

See the next section for details of how to generate the documentation for additional packages you install yourself.

Exercise 5.2: Read all about it

Find and view (or print) the documentation on the geometry package you used in Exercise 1.

Investigate some of the other package documentation files in the directory.

5.2 Downloading and installing packages

Once you have identified a package you need and haven't already got (or you have got it and need to update it) — and you're not using a system with the *tlmgr* auto-installer — you can use the indexes on any CTAN server to find the package you need and the directory where it can be downloaded from.

5.2.1 Downloading packages

What you need to look for is almost always *two files*, one ending in `.dtx` and the other in `.ins`. The first is a DOCTEX file, which combines the package programs and their documentation in a single file. The second is the installation routine (much smaller). You *must always* download *both* files. In most cases, CTAN provides a 'Download ZIP File' link, which gets you all the files you need for the package.

If the two files or the package are not there, it means one of two things:

□ *Either* the package is part of a much larger bundle which you shouldn't normally update unless you change version of LaTeX;[2]

□ *or* it's one of a few rare or unusual packages still supplied as a single `.sty` or `.cls` file originally written for the now obsolete LaTeX 2.09, or by an author who has a moral or philiosophical objection to using DOCTeX.[3]

Download both files (or the ZIP file) to a *temporary directory*. If you use Windows, create a folder like `C:\tmp` or `C:\temp` for this; Mac and Linux systems already have a `/tmp` directory. If you downloaded the ZIP file, unzip it there.

5.2.2 Installing a package

There are four steps to installing a LaTeX package:

1. **Extract the class or package files**
   Run LaTeX on the `.ins` file. That is, open the file in your editor and process it as if it were a LaTeX document (which it is), or if you prefer, type `latex` followed by the `.ins` filename in a command window in your temporary directory.

   This will extract all the files needed from the `.dtx` file (which is why you must have both of them present in the temporary directory). Note down or print the names of the files created if there are a lot of them (read the log file if you want to see their names again).

2. **Create the documentation**
   Run LaTeX or *pdfLaTeX* on the `.dtx` file *twice*. This will create a `.dvi` or `.pdf` file of documentation explaining what the package is for and how to use it. Two passes through LaTeX are needed in order to resolve any internal crossreferences in the text (a feature we'll come onto later). View or print this file in the usual manner (see § 4.3).

[2] For example, there is no `xcolor.dtx` and `xcolor.ins` for the xcolor package because it forms part of the graphics bundle, which is installed on all LaTeX systems anyway. Such packages change very rarely, as they form part of the core of LaTeX and are very stable. You should never try to update these packages in isolation.

[3] Almost all of these have been updated to work with LaTeX 2_ε, so they should be installed as in step 3 in the procedure on p. 82.

3. **Install the files**
   Move the files created in step 1 from your temporary directory to the right place[s] in your personal TEX directory tree (see below) — *always* your 'personal' TEX directory tree, *a*) to prevent your new package accidentally overwriting master files in the main TEX directories; and *b*) to avoid your newly-installed files being overwritten when you next update your version of TEX. Never, *never* put files into the TEX master installation tree. If you are updating a shared system, however, you can put files into the *local* TEX directory tree.

   'The right place' sometimes causes confusion, especially if your TEX installation is old or does not conform to the TEX Directory Structure. For a TDS-conformant system, 'the right place' is your personal TEX directory tree unless you are a systems manager updating a shared machine, in which case it's the local TEX directory tree. Your personal TEX directory tree is in your home directory (folder):

   □ Unix and GNU/Linux systems: `~/texmf/`

   □ Apple Mac systems: `~/Library/texmf`

   □ Windows MIKTEX systems: `Computer/username/texmf`

   Create this directory now if it does not already exist.

   Windows MIKTEX users (only) must use the MIKTEX Administration program to add the new folder to the search tree. Each time you update files in there, you must run the FNDB updater in the MIKTEX Administration program.

   'Suitably-named' means sensible and meaningful (and probably short). For a package like paralist, for example, I'd call the directory `~/texmf/tex/latex/paralist`.

   Often there is just a `.sty` file to move but in the case of complex packages there may be more, and they may belong in different locations. For example, new BIBTEX packages or font packages will typically have several files to install. This is why it is important to create a subdirectory for the package rather than dump the files into `misc` along with other unrelated stuff.

   If there are configuration or other files, read the documentation to find out if there is a special or preferred location to move them to.

4. **Shared systems and MIKTEX: update your index**
   If you are updating a shared system and putting the files into the local TEX directory tree, or if you are using Windows MIKTEX, you must afterwards run your TEX indexer program to update the package database. This program comes with every modern version of TEX and is

Table 5.1: Where to put files from packages

Type	Directory in your personal texmf/	Description
.cls	tex/latex/*packagename*	Document class file
.sty	tex/latex/*packagename*	Style file: the normal package content
.bst	bibtex/bst/*packagename*	BibTeX style
.mf	fonts/source/public/*typeface*	METAFONT outline
.fd	tex/latex/mfnfss	Font Definition files for METAFONT fonts
.fd	tex/latex/psnfss	Font Definition files for PostScript Type 1 fonts
.pfb	fonts/type1/*foundry*/*typeface*	PostScript Type 1 outline
.afm	fonts/afm/*foundry*/*typeface*	Adobe Font Metrics for Type 1 fonts
.tfm	fonts/tfm/*foundry*/*typeface*	TeX Font Metrics for METAFONT and Type 1 fonts
.vf	fonts/vf/*foundry*/*typeface*	TeX virtual fonts
.dvi	doc/*packagename*	package documentation
.pdf	doc/*packagename*	package documentation
others	tex/latex/*packagename*	other types of file unless instructed otherwise

variously called *texhash*, *mktexlsr*, or even *configure*, or it might just be a mouse click on a button or menu in your configuration system (like MIKTeX's). Read the documentation that came with your installation to find out which it is.

On Linux, GNU/Unix, and Apple Mac systems *do not* create an ls-R database in your ~/texmf or ~/Library/texmf directory or run the indexer. These systems search your personal TeX directory automatically.

On MIKTeX and shared systems, run your TeX indexer program after making changes

⚠

This last step is utterly essential, otherwise nothing will work.

Exercise 5.3: Install a package

Download and install the paralist package (which implements inline lists).

The *tlmgr* auto-updater is widely used in TEX Live systems *except* where TEX has been installed from Debian-based Unix *system* packages. On Windows and Apple Mac, and on Unix systems where TEX Live has been installed from the TUG DVD or download, *tlmgr* is the normal way to update packages. The manual process described above is *only* for those cases where *tlmgr* cannot be used.

This includes the thousands of installations which do not conform to the TDS, such as old shared Unix systems and some Microsoft Windows systems, so there is no way for an installation program to guess where to put the files: *you* have to know this yourself. There are also systems where the owner, user, or installer has chosen *not* to follow the recommended TDS directory structure, or is unable to do so for political or security reasons (such as a shared system where she cannot write to a protected directory).

The reason for having the local `texmf` directory (called `texmf-local` or `texmf.local` on some systems) is to provide a place for local modifications on a shared or managed system (such as a server) which will override but otherwise not interfere with the main TEX installation directory tree. Your installation should already be configured to look in the personal and local directories first, so that any updates to standard packages will be found there *before* the copies in the main `texmf` tree. All modern TEX installations do this, but if not, you can edit `texmf/web2c/texmf.cnf` (or on a shared system, ask your systems manager or support person to do so). There is an example in Appendix A.

5.2.3 Replicating the TDS

The TEX Directory Structure (TDS) is documented at `http://www.tug.org/tds/`. I find it useful to make the subdirectory structure of your local `texmf` directory the same as that of the main installation `texmf` directory. Examine the subdirectories of `texmf/tex/latex/` for examples. For updates of packages which came with your LATEX distribution (as distinct from new ones you are adding yourself), you can then use the same subdirectory name and position in your local `texmf/...` as the original used in the main `texmf/...`, and LATEX will then always use the updated version.

If you want to create the entire subdirectory structure ready for use, you can do it under Unix with the following commands (this example uses the

Ubuntu/Debian directory /usr/local/share/texmf rather than the Red Hat /usr/share/texmf-local):

```
cd /usr/share/texmf
find . -type d -exec mkdir -p /usr/local/share/texmf/{} \;
```

If you are using Microsoft Windows, you can download *Cygwin*, which provides you with the standard Unix tools in a shell window. The above command also works on a Mac running OS X if you use /usr/local/texlive/*yyyy*/texmf-dist in the first command (replacing *yyyy* with the year of the distribution) and ~/Library/texmf/{} in the second. In other cases, if your directories are not as given in the example, you need to substitute the actual paths to your main texmf and local texmf directories.

5.3 Online help

The indexes and documentation files on CTAN are the primary online resource for self-help on specific packages, and you should read these carefully before asking questions about packages.

5.3.1 The FAQ

For general queries you should read the Frequently-Asked Questions (FAQ) document so that you avoid wasting online time asking about things for which there is already an easily-accessible answer.

The FAQ is managed by the UK TeX Users Group and can be found at http://www.tex.ac.uk/faq/.

5.3.2 The TeXhax mailing list

Another support resource is the mailing list texhax@tug.org. Again, feel free to ask questions, but again, try to answer the question yourself first (and say what you've tried in your message).

5.3.3 Web sites

The TeX Users Group, as well as most local user groups, maintains a web site (http://www.tug.org) with lots of information about various aspects of the TeX system. See Appendix B for information on joining TUG.

5.3.4 News

The Usenet newsgroup `comp.text.tex` is the principal forum for other questions and answers about LaTeX. Feel free to ask questions, but please do not ask frequently-asked questions: read the FAQ instead. The people who answer the questions do so voluntarily, unpaid, and in their own time, so please don't treat this as a commercial support service.

To access Usenet news, type the following URI into your browser's 'Location' or 'Address' window: `news:comp.text.tex` (if your browser doesn't support Usenet news properly, change it for one that does, like *Mozilla*[4]), or download one of the many free newsreaders.[5]

5.3.5 Google LaTeX list

There is a Google Groups mailing list for LaTeX users at `http://groups.google.com/group/latexusersgroup?hl=en`

5.3.6 Commercial support

If you need commercial levels of support, such as 24-hour phone contact, or macro-writing services, you can buy one of the several excellent commercial versions of TeX, or contact a consultancy which deals with TeX (details on the TUG Web site).

[4] `http://www.mozilla.org/`

[5] Note that this means newsreaders for the Usenet News (NNTP) service. It does *not* mean syndication readers for RSS, which are a different thing entirely — these are unfortunately also sometimes referred to as 'newsreaders'.