A \LaTeX{} Tour, part 2:  
The Tools and Graphics distributions  
David Carlisle

Introduction

In the previous article in this series I started by giving a description of the files in the 'base' \LaTeX{} distribution. In part 2, I shall cover the 'tools' and 'graphics' distributions. These are distributed in the \texttt{tools} and \texttt{graphics} subdirectories of the CTAN directory \texttt{macros/latex/packages}. Although these files are not part of the minimal base distribution they should normally be included in the \LaTeX{} installation at any site. The \LaTeX{} book assumes that at least the graphics distribution is installed.

The primary source for \LaTeX{} is the CTAN\footnote{ftp.tex.ac.uk in the UK, ftp.dante.de in Germany, and ftp.cdrom.com in the US} network of archives, so if I refer to path names of files this relates to the CTAN file structure. Note however that if you obtained \LaTeX{} as part of a pre-packaged \TeX{} distribution, then these files may have been moved (typically documentation files may be separated from \TeX{} source files). I hope this will not cause any confusion.

The Tools Distribution

The tools distribution consists of packages written by individual members of the \LaTeX{}3 project. They are supported by the same mechanism as the base \LaTeX{} distribution, that is, any problems should be reported using \texttt{latexbug.tex} and the \LaTeX{} bug report database, as described in part 1. Note that this bug report system should not be used for 'contributed' packages that one may find in the \texttt{macros/latex/contrib} area of the CTAN archives.

Packages Extending the array and tabular Environments

The first group of packages extends the functionality of the standard \LaTeX{} array and tabular environments. These are all described in Chapter 5 of \emph{The \LaTeX{} Companion}, as well of course as in the source '.dtx' files which may be processed by \LaTeX{} to produce typeset documentation, and optionally code listings.

\texttt{array} Extended versions of the \texttt{array}, \texttt{tabular} and \texttt{tabular*} environments. The principal advantage of the versions provided by this package is that you can specify typesetting instructions to apply to a whole column of the table. As well as the usual \texttt{clr} column specifiers, one may add commands at the beginning of each entry with \texttt{>} and at the end of each entry with \texttt{<}. So a column specifier of \texttt{>{\bfseries}>c{\bfseries}} would produce a bold, centred column of a table.

The \texttt{array} package also provides a command \texttt{\newcolumntype} for defining new column specifiers, in addition to the standard ones. This is used by some of the packages described below.

\texttt{dcolumn} Alignment on 'decimal points' in tabular entries. Requires \texttt{array}. This package provides a new column specifier \texttt{D} which may be used to produce columns of numbers aligned on a decimal point '.', or some other symbol, such as ',', or '}'.

\texttt{delarray} This package requires the \texttt{array} package. It provides a mechanism for specifying 'large delimiters' around arrays. This is most convenient for putting brackets around arrays that are to be aligned on their top or bottom row (when the 'obvious' construction with \texttt{\left} and \texttt{\right} does not work). Compare the standard
\begin{verbatim}
\left( \begin{array}{t} a \hline b \end{array} \right)
\end{verbatim}
with the effect produced using the \texttt{delarray} syntax.
\begin{verbatim}
\begin{array}{cc}
\left( \begin{array}{c} a \hline b \end{array} \right) & \end{array}
\end{verbatim}

\texttt{hhline} Finer control over horizontal rules in tables. Requires \texttt{array}. Standard \LaTeX{}'s \texttt{\hline} construction produces a double rule across a table, but the user has no control over how this rule interacts with vertical rules. Using the \texttt{\hhline} command provided by this package gives 'corners' where a double horizontal rule meets a double vertical rule, and other similar effects.

Compare the first, standard, construction with the following \texttt{\hhline} sample:
\begin{verbatim}
\begin{array}{|c|c|}
\hline
\end{array}
\end{verbatim}
You must type a valid file name to this prompt or \TeX just repeats the request. On some systems you can use a mechanism to abort the job (e.g., control-C or control-Z) but there is no way to tell \TeX to skip the input or do any other error recovery.

To avoid this unpleasant loop, \TeX always checks that a file exists before trying to input it (unless you use the primitive \texttt{\textbackslash input filename} syntax with no \texttt{\{ \}} around the argument).

If you do encounter this loop using a different format, or with \LaTeX, by mistyping the file name on the command line, then the following .\texttt{tex} files provide valid filenames that you can easily remember to type at the missing file prompt. The actions that each of these .\texttt{tex} files takes is designed to mimic the actions that are possible after a \TeX error.

h Typing \texttt{h (return)} to the missing file prompt will cause \TeX to input \texttt{h.tex}. This produces a helpful message, and then produces the normal `error prompt' i.e., ? so you can hit (\texttt{return}) to move on, or \texttt{x} to quit, or whatever.

s Typing \texttt{s (return)} inputs \texttt{s.tex} which puts \TeX into 'scroll mode'. This means that it will scroll past future errors without stopping.

x Typing \texttt{x (return)} causes the current \TeX run to be aborted.

e The file \texttt{e.tex} is in fact the same as \texttt{x.tex} but allows \texttt{e} to be given as an answer to the missing file prompt similar to the \texttt{e} response to the error prompt (which is supposed to start up an editor but usually is the same as \texttt{x}).

u If the operating system allows there will also be a file \texttt{.tex} which does nothing, this will mean that just hitting (\texttt{return}) in response to the missing file prompt inputs .\texttt{tex} and allows \TeX to proceed with the original file. Some operating systems object to a file with only an extension and no filename before the '.', so this option may not be available to you. Most \TeX distributions include a file \texttt{null.tex} which is also empty, so if you do not have the option of installing the file .\texttt{tex} you may type \texttt{null (return)} in response to the missing file prompt, which will also allow \TeX to proceed.

**Miscellaneous Tools Packages**

\texttt{afterpage} Defines an \texttt{\textbackslash afterpage} command that saves up its argument and executes it after the current page (i.e., at the top of the next page). \LaTeX's output routine was not designed with the idea that packages might want to play this kind of trick, so this package is particularly fragile. In fact it was only written as a kind
of private joke; I noticed the comment "Output routines are always protected by enclosing them in groups, so that they do not inadvertently mess up the rest of \TeX" in the \TeX{}book, and wanted to answer\footnote{The answer incidentally is not at the top of the next page, but rather any of the places where the \TeX{}book uses the magic phrase "exercises the page builder".} the question, "Where do you end up if you jump out of that group with \verb|\aftergroup|?" Despite this, judging by reactions, people do seem to find the package useful...

\textbf{enumerate} Extended version of the enumerate environment. The environment is given an optional argument which controls how the counter is printed. For example \texttt{\begin{enumerate}[a]} would produce items labelled (a)' \ b)' \ c}'.

\textbf{ftnright} Place footnotes in the right hand column in two-column mode. Normally \LaTeX{} places footnotes at the bottom of each column. This package causes the footnotes for both columns of a page to be set in the normal text area at the end of the second column on each page. It currently works only with the standard two column mechanism, not with the mechanism of the multicol package.

\textbf{indentfirst} Indent the first paragraph of sections etc. This very small package just suppresses the usual \LaTeX{} mechanism which ensures that the first paragraph of each section is not indented.

\textbf{multicol} Typeset text in columns (up to 10 columns per page), with the length of the final columns 'balanced'. \textit{Baskerville} uses this package to balance the columns at the end of every article. Unlike the standard \texttt{\twocolumn} command, this package allows changing the number of columns part way down a page. It does have some restrictions on the use of floats which means that it is not suitable for all purposes. Also (uniquely for the files in the core \LaTeX{} distribution) this package has special conditions on commercial use.

\textbf{rawfonts} Preload fonts under the old internal font names of \LaTeX{} 2.09. Not recommended for new packages, but may help when updating old files.

\textbf{somedefs} This package is not intended to be called directly by a document, but may be used (via \texttt{\RequirePackage}) to build a package in which you want the default behaviour to be to execute \textit{all} possible options, but that the user may execute just some of the options by specifying options in the \texttt{\usepackage} call. This is used in the rawfonts package above to allow just some of the 'old' font names to be defined rather than all of them.

\textbf{theorem} Flexible definition of 'theorem-like' environments. The standard \texttt{\newtheorem} command gives some control over the title and numbering of 'theorem-like' declarations, but is not very flexible. The theorem package provides an enhanced declaration scheme which gives control over the fonts used in the heading and theorem body, and such details as whether the numbering is 'Theorem 1' or '1 Theorem'. Recently this package has acquired a close cousin, the amsthm package, part of the 'AMS-\LaTeX{}' collection. The AMS variant has perhaps slightly simpler user-syntx but is used in much the same way.

\textbf{varioref} Provides \texttt{\vref} and related commands. \texttt{\vref} is like a combination of \texttt{\ref} and \texttt{\pageref} which produces references such as 'Figure 2 on page 3'. However, it omits the page number if it is on the current page, and replaces it by phrases such as 'on the facing page' when appropriate.

\textbf{verbatim} Flexible version of verbatim environment. The standard \LaTeX{} verbatim environment can not easily be used in the definition of other environments as typically the \texttt{\end} of the newly defined environment is not recognised as such, but is treated as verbatim text. This package re-implements verbatim such that (with some restrictions) it can be used in the definition of other environments and commands. It also defines some such derived environments, for inputting and writing files verbatim, and for adding line numbers, and also a comment environment that ignores all the environment body.

\textbf{xr} The xr (\textit{external} \texttt{references}) package allows one \LaTeX{} document to access the .\texttt{aux} file of another. So if fileA has a section marked with \texttt{\label{xyz}} then fileB may refer to that section using \texttt{\ref{xyz}} just as if it were part of the same document. This requires the file fileA.\texttt{aux} created when fileA was processed to be still available when fileB is processed. (This package was originally by Jean-Pierre Drucbert, but was recoded and adopted into the tools distribution.)

\textbf{xspace} One of the more common errors in \LaTeX{} documents is to use a command such as \texttt{\TeX} within text, but forget to follow it with \texttt{\\} or \texttt{\{}. This package defines a command \texttt{\Xspace} which may be used at the end of the definition of such a 'text command'. It looks ahead
and adds a space unless the next token is a punctuation character.

Packages for Drafts and Tests

\fontsmpl Package and test file for producing `font samples`. The base distribution contains a file \texttt{nffont.tex} that shows some small samples, and a character table for a given font. \texttt{fontsmpl.tex} produces a much more extensive test showing examples of all the fonts in a given family. If you want to devise your own similar test suite you may use the \texttt{fontsmpl} package, following the examples in \texttt{fontsmpl.tex}.

\texttt{layout} Defines a `layout' command that produces a half size `picture' of the document page settings such as \texttt{textwidth, oddsidemargin,...} together with a table of their values. This is quite useful when designing a new class file, as it gives a visual representation of how the various areas of the page for headings, body text, marginal notes, etc., relate to each other.

\texttt{showkeys} \LaTeX{}'s automatic numbering and cross referencing feature is one of its strongest points, as it makes editing a document (and thus potentially changing the numbering throughout the rest of the file) quite painless. However, one disadvantage is that when reading a printed draft, one sees `final' numbers rather than the symbolic names that are used in the source file's \texttt{label} and \texttt{ref} command. This package makes these symbolic names, or `keys', appear in the margin in the case of \texttt{label} and \texttt{bibitem} or raised above the number, like this: \texttt{...} in the case of \texttt{ref} and \texttt{cite}. Some people find the raised labels above cross references distracting and so a package option turns them off, just leaving the marginal notes showing the \texttt{label} and \texttt{bibitem} keys.

The Graphics Distribution

\LaTeX{} (and the \texttt{dvifile}) is only designed to deal with rectangular boxes consisting of text, white space or rectangular rules. However it has an `escape mechanism', the \texttt{special} primitive command that allows processing instructions to be passed straight from \LaTeX{} (via the \texttt{dvifile}) to the `driver' program that is used to process (e.g., preview or print) the \texttt{dvifile}. \LaTeX{} places essentially no restrictions on what instructions may be passed via \texttt{special}, and so the possibilities are unlimited...

Most modern drivers can import `graphic' files of various sorts. Those drivers that are producing PostScript can often do more extensive manipulations of the typeset text, such as scaling or rotation of the text, or even writing text along an arbitrary curve. Many of these drivers can also support colour to some extent. Unfortunately as all these features require that the \texttt{dvifile} stores processing instructions for the driver, it means that the \texttt{dvifile} is not portable to a site that uses a different driver program. There have been many attempts over the years to coordinate the \texttt{special} syntax used by the different drivers, so that they would all accept a common core of processing instructions, but there has been notable lack of success in such efforts to date...

As a `next best thing' to having portability at the \texttt{dvifile} level, \LaTeX{} supplies a suite of standard graphics commands provided by the packages described in this section, so that at least \LaTeX{} source files should be reasonably portable. At a given site the graphics packages will be customised to use a suitable `back end' file that converts the \LaTeX{} syntax into the form required by the local driver. This should mean that as long as both drivers support some feature, such as including PostScript graphics, a file just needs to be re-processed with \LaTeX{} to use the \texttt{specials} at the new site; the \LaTeX{} file does not need to be edited. Although this suite of programs was devised as part of \LaTeX{}, users of other \TeX{} formats may use them by way of the interface available from CTAN hosts in macros/generic/graphics.

Documentation All the packages in this distribution are, as usual, distributed as documented sources in \texttt{dtx} form, however the documentation in these package sources is rather technical. A separate `User Guide' is available as \LaTeX{} source in \texttt{grfguide.tex} and also in pre-formatted form in the PostScript file \texttt{grfguide.ps}.

The color package (which produces colours despite the strange spelling) and the graphics package are also described in Lamport's \LaTeX{} manual. An alternative to \texttt{grfguide.tex} as a free source of documentation is Keith Reckdahl's \textit{Using EPS Graphics in \LaTeX}2e Documents distributed from CTAN sites in the file \texttt{info/epslatex.ps} and published in \textit{TUGboat} 17, no. 1–2. This document covers the \texttt{graphicx} package in some depth, and also related contributed packages for controlling figure placement and captions, and the \texttt{psfrag} system for overlaying \LaTeX{} text over a PostScript diagram.

Colour

\texttt{color} Produce coloured effects in your document.

The \texttt{color(red)} declaration would make all the following text red, the similar \texttt{textcolor}
command takes an extra argument that specifies the text to be coloured (by analogy with \textfamily and \textrm).

One may also produce boxes with coloured backgrounds using the \textcolorbox command.

Accurate treatment of colour is probably the feature that requires the most 'help' from the driver program. If your driver was not specifically written to support colour then probably the color package will not work at all, or will be limited to regions of colour that fall on one page, and all the current colours will be 'forgotten' at a page break.

\textsc{pstricks} The \textsc{pstricks} package of Tim van Zandt provides a very powerful interface to PostScript. Unfortunately the package has some slight incompatibilities with the color package. If a document loads pstricks, both color and \textsc{pstricks} are loaded, and then a few internal \textsc{pstricks} functions are redefined to repair the incompatibility.

\textbf{Rotation, Scaling and Graphics Inclusion}

\texttt{graphics} This is the core \LaTeX{} package for rotation (\rotatebox), scaling (\scalebox and \resizebox) of text, and the inclusion of graphics images (\includegraphics). Unlike the old \TeX{} packages such as psfig the \includegraphics command is not restricted to PostScript graphics, but can include any graphics formats that your driver supports.

\texttt{graphicx} Often when including graphics files one needs to specify combinations of scaling and rotation and other special effects. The \texttt{graphicx} package uses standard \LaTeX{} 'positional' optional arguments which means that it is not practical for any command to support more than a couple of optional arguments. The \texttt{graphicx} package calls the \texttt{graphics} package internally, but offers a more powerful and friendly 'named argument' interface in which an arbitrary number of optional keys may be set in one \{ \} argument. For instance to include a graphics scaled to half size, and rotated through 90°, one can specify

\begin{verbatim}
\includegraphics[scale=.5, angle=90]{file}
\end{verbatim}

To do the equivalent with the \texttt{graphics} package would require nested calls of \includegraphics inside \scalebox inside \rotatebox.

\texttt{lscape} Provides a landscape environment within which the body of every page is rotated through 90°. The page head and foot are not rotated, but stay in their usual positions. It requires the \texttt{graphics} package which is used to handle the rotation.

\texttt{epsfig} The obsolete \LaTeX{} 2.09 did not come with a standard graphics package. Two popular contributed packages to include PostScript graphics were \texttt{psfig} (T. Darrell) and \texttt{epsf} (T. Rokicki). Sebastian Rahtz merged and extended these to produce the package \texttt{epsfig}. The \texttt{epsfig} package became very popular, especially after it was given extensive coverage in \textit{The \LaTeX{} Companion}. For this reason the current distribution contains a package called \texttt{epsfig} so that old documents do not need converting to the new system. However this \texttt{epsfig} is just a wrapper that converts the old syntax into calls to the new \texttt{\includegraphics} command and so should not now be used for new documents.

\textbf{Driver Files} As mentioned above, these packages all require customisation to a particular driver. This may be specified either in a site configuration file, or as a package option in the document. The code for these drivers is all stored in '.def' files, so for instance the code for the \dvips driver (and also for \xdvi which uses the same \texttt{\special} syntax) is stored in \dvips.def. All these driver files are derived from the same source, \texttt{drivers.dtx}, except for the \texttt{Textures} file which is currently distributed as a separate \texttt{textures.def} contributed by Arthur Ogawa. One special .def file does not correspond to a driver, \texttt{dvipsnam.def} predefines the 60+ colours that are 'known' to the \dvips driver. It may be used with other drivers as well, as described in the \texttt{color} package documentation.

\textbf{Other Graphics Packages} The remaining two packages do not have code that is specific to \dvips driver programs, and so in some sense do not really belong in the graphics distribution; they are used by the \texttt{graphics} and \texttt{graphicx} packages. In fact the code of either of these packages may be extracted and used in any format based on plain \TeX{}. They do not use any \LaTeX{} specific features.

\texttt{keyval} The \texttt{graphicx} package makes use of a 'named argument' or 'key equals value' syntax as described above. Keyval provides a general parser for such a syntax, so this package is unlikely to be directly called within a document, but may be loaded by \texttt{\RequirePackage} by any package or class file that needs to define commands with such a syntax.

\texttt{trig} This package provides functions for calculating the trigonometric functions \sin, \cos and \tan. These are used by the \texttt{graphicx} package for determining the amount of space a rotated box will take up.
Coming Soon

Part 3 of this tour will describe the files of Johannes Braams' \texttt{babel} distribution of packages for multi-lingual typesetting, the \texttt{psnfss} distribution of Post-Script font related packages, and \texttt{mfnfss} distribution of packages for loading 'Pandora' and 'Old German' fonts.

\@ David Carlisle
Mathematics Department
Manchester University
Oxford Road
Manchester, England M13 9PL UK
carlisle@ma.man.ac.uk