Travels in \TeX{} Land: Tweaking \LaTeX{} to do something a little different

David Walden

1 Introduction

Sooner or later, many users new to \LaTeX{} begin to want to tweak \LaTeX{} to work differently than it does “out of the box.” These tweaks may be

- to provide capabilities that do not already exist in \LaTeX{}
- to modify existing \LaTeX{} capabilities to work a little differently
- to give \LaTeX{} a considerably different look—perhaps one that feels more comfortable to the user or more appropriate for the user’s task
- for a combination of these or other reasons

Such changes are possible by changing existing \LaTeX{} parameters (e.g., counters and lengths), by creating user-defined commands and environments, and by modifying existing \LaTeX{} capabilities.

The following sections of this note sketch several ways of making such changes, so relative beginners can get a picture of some of the spectrum of possibilities:

- finding an appropriate package or class (Section 2)
- creating a new command or environment to do something new (Section 3)
- creating a new command or environment to do something differently than \LaTeX{} already does it (Section 4)
- changing an existing command or environment (Section 5)

Note that I, too, am far from being an expert \LaTeX{} user. This note sketches some of what I would have liked to have found sketched in one place (rather than having to hunt in books or on the Web) when I was first trying to tweak \LaTeX{}.

\footnote{Another good place to start is chapter 10 of Helmut Kopka and Patrick W. Daly’s book \textit{A Guide to \TeX{}}, fourth edition, Addison-Wesley, 2004.}
2 Finding an appropriate package or class

One approach to getting \LaTeX{} to perform differently than it does “out of the box” is to find an existing class, package, etc.,\(^2\) that supports what you want. For instance, many people want tighter spacing for list structures than basic \LaTeX{} provides, and you can find a number of classes and packages at

\url{http://www.tex.ac.uk/faq}

that change various aspects of lists.

However, in this note I will mostly ignore the option of finding another class or package to do what you want and will concentrate on modifying the class or package you are already using. First, it can be a bother to be shifting among classes depending on what behavior you want from \LaTeX{} for a particular document because then you have to be aware of subtle differences in how things work from document to document. Second, there are lots of existing \LaTeX{} capabilities (classes, packages, etc.), and finding the one you want could be a big project. However, if you want to search for a class or package to address your particular need, here are some places to look:

- the \TeX{} FAQ (\url{http://www.tex.ac.uk/cgi-bin/texfaq2html?introduction=yes})
- one of the CTAN search engines at \url{http://www.tug.org/ctan.html}
- the \texttt{comp.text.tex} Usenet group
- \textit{The \LaTeX{} Companion} (Frank Mittelbach et al., 2nd ed., Addison-Wesley, 2004)

3 Doing something new

Sometimes you want to extend \LaTeX{} to do something completely new. Of course, creating new commands and environments is covered in many books on \LaTeX{}. See, for instance, sections 10.3 and 10.4 of Kopka and Daly’s book — cited in footnote 1. Nonetheless, I will give three simple examples. For instance, I always have the \texttt{\CK{}} command defined that produces a boldface \textbf{CK} to mark places where something needs checking in a manuscript I am drafting. My trivial definition for this command is

\begin{verbatim}
\newcommand{\CK}{\textbf{CK}}
\end{verbatim}

Thus, if I give this command after a fact I want to remember to check, e.g., a date as in

\texttt{was born in 1948\CK{}}

the printed result will be

\texttt{was born in 1948\CK{}}

\(^2\)Generally, speaking, classes, which come in \texttt{.cls} files, specify the overall structure and look of a document; generally speaking, packages, which come in \texttt{.sty} files, augment or extend basic \LaTeX{} or classes.
Here is another example of a new command to extend the capability of \LaTeX. When I began writing a book using \LaTeX,\textsuperscript{3} I wrote a set of commands to insert figures and tables, either numbered or unnumbered (for in-line insertion in the body of the text), and scaled or not scaled. Figure 1 is my command for inserting a scaled, numbered .\texttt{eps} figure into my book.\textsuperscript{4}

\begin{verbatim}
\newcommand{\snfig}[3]{ %insert a scaled numbered figure
 \begin{figure}[htbp]
 %center it and scale the included .\texttt{eps} figure
 \hfill\scalebox{#3}{\includegraphics{fig#2.eps}}\hfill
 %must have done \usepackage[dvips]{graphics} to use \includegraphics
 \caption{\label{fig:#2}#1 \texttt{\small[fig#2]}}
 \end{figure}
}
\end{verbatim}

Figure 1. A new command, \texttt{/snfig}

It is called with three arguments, e.g.,

\texttt{\snfig{caption}{unique part of fig’s file name}{scale factor}}

In addition to inserting the .\texttt{eps} figure, the command creates a label by which the figure can be referenced and puts the figure file id at the end of the caption. The latter is deleted from the definition before printing the finished manuscript, but it is very handy to have while continuing to edit the figures and the manuscript. For instance, the following command

\texttt{\snfig{Schematic of contingency planning}{060}{.4}}

results in the output in Figure 2, assuming I previously created the figure in the file fig060.\texttt{eps}.

From time to time I have also added new functionality to \LaTeX using a new \texttt{env}ironment. See, for instance, Figure 3, an environment I use to write distinctive notes to myself in the manuscript I am drafting.

Authors’ note(s): Here is an example of the result of using the \texttt{anote} \texttt{environment}.

Much of the time, however, you want to change the way an existing aspect of \LaTeX works. I describe some ways to do this in the following sections.


\textsuperscript{4}For more about how figures and tables work, see chapter 9 of Kopka and Daly’s book cited in footnote 1.
4 Doing things differently

As mentioned in Section 2, often people want tighter spacing within a list than the itemize, enumerate, and description environments normally provide. When I first started using \LaTeX to write my book, I tried to change the parameters of the built-in list environments to tighten spacing, but I couldn’t make incrementing and decrementing the various list parameters work correctly.\footnote{Changing an existing aspect of \LaTeX}{Therefore, I wrote my own environments with absolute values for the various parameters as shown, for example, in Figure 4.\footnote{More about this in Section 5.4.}}

5 Changing an existing aspect of \LaTeX

5.1 Changing parameters for an existing part of \LaTeX

Lots of aspects of \LaTeX can be modified by changing length and counter parameters.\footnote{See figure 4.1 on page 76 of Kopka and Daly’s book cited in footnote 1 for definitions of the various length parameters relating to lists. I also followed Kopka and Daly’s examples (section 4.4.4 of their book) for how to create a list environment. My brute force approach of just changing the length parameters until I like how things looked lacks generality in more ways than one. Why changing parameters doesn’t work to change the function of the list environments is explained at the URL given at the end of the first paragraph of Section 2.}{For instance, you can change the amount new paragraphs normally indent with a command such as

\begin{verbatim}
  \setlength{\parindent}{.5in}
\end{verbatim}

\footnote{See sections 10.1 and 10.2 of Kopka and Daly’s book for a list of such lengths and counters.}
%define the blist environment for bulleted lists with less
% space between items than in the itemize environment
\newenvironment{blist}{\begin{list} %stuff for before my list items
{\bfseries\upshape $\bullet$}
 {\setlength{\labelwidth}{.1in} \setlength{\leftmargin}{.35in}
 \setlength{\labelsep}{.1in} \setlength{\rightmargin}{0cm}
 \setlength{\parsep}{2pt} \setlength{\topsep}{4pt}
 \setlength{\itemsep}{-2pt}
}} %end of before-stuff
{\end{list}} %stuff for after my list items

Figure 4. A new environment, blist

5.2 Redefining an existing part of \LaTeX

Rather than defining the new blist environment as I did in Figure 4, I could have redefined the itemize environment itself. To do this, I would use

\renewenvironment{itemize}

rather than

\newenvironment{blist}

in Figure 4 and also change the introductory comment appropriately.

Giving the modified capability a new name has the advantage that you can call either the normal or modified version in your document. Modifying the normal version and keeping its name has the advantage that you can change the action of this capability in your document without going back to change the name where it is used throughout your document.\footnote{The method of modifying the itemize environment described here throws away much of the generality of itemize as built into \LaTeX.}

5.3 Modifying an existing part of \LaTeX

For the book I was writing when I began to learn \LaTeX, I wanted figure and table captions to use a period rather than a colon after the figure or table number, and I wanted both the table or figure number and the caption itself to be in bold face type. I did this as shown in Figure 5.

I learned how to make these changes by querying the comp.text.tex list where I was told to find the appropriate part of the latex.ltx file (which includes lots of \TeX) in my \TeX distribution, copy that part to the preamble of my document, and make the desired change to the \LaTeX basic definition now in my preamble, which would redefine this aspect of basic \LaTeX. I didn’t really understand the \TeX but I was
able to accomplish my purpose by adding the `\textbf` commands around arguments #1 and #2 in lines 5 and 7 of the figure and also adding the period in both lines. Sometimes, especially if the change required is to a package other than basic \LaTeX, I make a private copy of the package giving it a file name that makes it clear that this is not the official version and specify that with my `\usepackage` command. This is discouraged in the \LaTeX literature for good reasons, I am sure. However, oftentimes I find it more convenient to change a private copy of a package rather than making a change in my preamble as shown in Figure 5, and I don’t see much problem with it as long as I rename the private copy of the package.

5.4 Adding to an existing part of \LaTeX

Suppose for some reason I wanted to set off list environments with horizontal rules. Of course, I could define a new list environment that included these horizontal rules. For instance, to do this I could change Figure 4 to have

\begin{verbatim}
\newenvironment{hr-blist}{\hrule\begin{list}}{\end{list}}
\end{verbatim}

in place of

\begin{verbatim}
\newenvironment{blist}\begin{list}\end{list}
\end{verbatim}

I would also add `\hrule` immediately after `\end{list}` in the last line of Figure 4 and appropriately change the introductory comment.

However, if I wanted all instances of use of the `itemize` environment to include such horizontal rules without me having to change the name of each use, I could redefine the existing version that mostly uses the normal version but with my modification. I also learned how to do this from `comp.text.tex`, as shown in Figure 6,
which required going to the file for basic \LaTeX (latex.ltx) to see how the itemize environment is defined.

There is a significant difference between how I handled the change in Figure 5 and how I handled it in Figure 6. In the former case, you have to copy all the ugly code for handling captions into your program. In the latter case, you only have to know the necessary “hooks” into the ugly code. Unfortunately, in either case, you have to look at the ugly code and understand it enough to make changes or hook into it — unless some nice person just gives you the verbatim change you want, as happened to me in both instances.

Acknowledgments

Karl Berry helped improve the presentation and accuracy of this note, as did the journal’s reviewers. I especially appreciate the authors of various books in which I looked and of the people on the comp.text.tex list who answered my questions — they taught me everything I know related to this article.

Biographical note

David Walden is retired after a career as an engineer, engineering manager, and general manager involved with research and development of computer and other high tech systems. Readers with suggestions relating to this paper can contact him via his web site, www.walden-family.com.