The PracTEX Journal
TPJ Vol 1 No 1, 2005-01-15
Rev. 2005-01-24

\begin{here}

Doug Waud and Tim Null

1 Introduction

This is the first in a series of columns designed for the non-expert TEX and LATEX user or wannabe. This column might be for you. Perhaps you have not had the opportunity, inclination, or courage to fully develop your TEX-related skills. If you are dissatisfied with your TEX-skills, and you are looking for an opportunity to improve them, then we invite you to \begin{here}, and set out on a quest for greater TEX knowledge and expertise[1].

We caution you not to expect the answers to all your questions in this first column. After all, at this point, we don't even know your questions, but that's one reason we really want to hear from you. If you'll pass on your questions and concerns, it will help us develop topics for future columns. If you have tips, tricks, and suggestions, please pass those along, too. We may be able to add some of those to future columns as well.

This column is divided into two main parts: (1) The Installation of TEX, and (2) Project 1. We begin with the installation of TEX because in our early days with LATEX we were always frustrated and annoyed by documentation that began with "We will assume you already have TEX installed on your system." However, we won't give you a complete "howto" on the installation process. What we will do is give you an overview, and a good listing of resources where you can go to obtain TEX, and learn about installation. We will also give you some ideas on what you can do to learn more about using TEX and LATEX. Also, we couldn't resist the temptation to

[1]The original source file for this paper can be downloaded at http://www.tug.org/pracjourn/2005-1/waud/waud.tex.

give you a little taste of LaTeX, so this first column contains a simple LaTeX project. All this makes this first column a little long, but we hope you will find the column readable and well worth the effort.

1.1 *TUGboat* Beginner's Article

At the same time this column is being launched, a parallel introductory column is being added to *TUGboat*. *TUGboat* is a publication of the TeX Users Group (TUG). The *TUGboat* article, *What is TeX?*, will be published in Volume 24 issue Number 2 of *TUGboat*. The article begins on page 163 of that issue. Both venues are intended to help less able TeX and LaTeX users develop confidence and skill. The first *TUGboat* article gives a summary of what TeX and LaTeX are all about, and we shall not repeat that history here. If you do not have access to the printed version it may be found in the *TUGboat* archive. Look online at the following address for the second issue in the year 2003:

http://www.tug.org/tugboat/contents.html

1.2 How to Read this Article

As we move along we will be mentioning information relevant for different operating systems. Some of the information will be specific to Windows®, some will apply only to the Mac®, still other stuff will be related to Linux®. This could get a little confusing if you attempt to read and comprehend everything; for example, if you're a Mac user, and you try to apply the Windows stuff to OS X.

We suggest that you just skip over stuff that applies to operating systems you don't use.

2 Installing TeX and LaTeX

2.1 First Step: Documentation

You will need information about TeX and LaTeX commands, and how to set up your TeX installation. The first thing you will probably want to do, if you haven't already, is visit the "New to TeX" area at the TeX User's Group web site. This

area includes "Getting started," FAQ, and "History of TEX" sections. The "Getting started" section is an obvious place for new TEX users to start:

http://www.tug.org/begin.html

As long as you are visiting the TEX User's group web site, you should take a few moments, and become a user group member.

Information on TEX User's Group membership is available on this web page:

http://www.tug.org/join.html

As a TEX Users Group member you will receive the printed journal *TUGboat*, and annual CDs or DVDs with updated TEX stuff on them—a great convenience. To find out other advantages and benefits of TEX Users Group membership, please check out the list on this web page:

http://www.tug.org/aims_ben.html

2.1.1 Free Documentation

A list of free documentation that can be downloaded is available at the TEX Users Group web site on the "Getting started" page listed above. We recommend you download the Tobias Oetiker PDF, *The Not So Short Introduction to LATEX2e*, for its excellent introduction and its table of mathematical symbols.

The Not So Short Introduction to LATEX2e can be downloaded at:

http://www.tug.org/tex-archive/info/lshort/english/lshort.pdf

Formatting Information: A Beginner's Introduction to Typesetting with LATEX, written by Peter Flynn, is another good document to have in your library. It covers a comprehensive range of topics related to LATEX formatting, and it is an excellent starting point for almost any LATEX-related topic. A copy of this document can be downloaded at:

http://www.tug.org/tex-archive/info/beginlatex/

Also, there are several useful sources of online information listed on the TUG "Getting starting" page. In addition, you can visit the TEX Frequently Asked Questions page at

http://www.tex.ac.uk/faq ,

you can visit the LaTeX Navigator web site

http://tex.loria.fr/english/,

or you can always do a Google search for "latex help tutorial" (or something similar), and you will get a listing of numerous online resources.

2.1.2 A Trip to the TUG Bookstore

There are numerous excellent books about LaTeX. We will mention only two. Both these books come with a CD that contains an abbreviated version of the LaTeX system contained in the TeX Live 2003 CD—Windows, Mac, and Linux (but not other Unix implementations). (We will briefly discuss the TeX Live 2003 CD below.) These CDs also contain examples and demos relevant to the respective book.

Helmut Kopka and Patrick W. Daly. (2004). *Guide to LaTeX* (4th ed.). Boston, MA: Addison-Wesley Professional.

Frank Mittelbach and Michel Goossens with Johannes Braams, David Carlisle, and Chris Rowley. (2004). *The LaTeX Companion* (2nd ed.). Boston, MA: Addison-Wesley Professional.

Both these books are frequently available for purchase at local booksellers. They can also be purchased online; for example, at the TUG web site:

http://www.tug.org/books/

They are both excellent books. The Kopka and Daly book is updated more frequently than *The LaTeX Companion*, so if you're on a tight budget, you might consider purchasing *The LaTeX Companion* now, and then get the next edition of the *Guide to LaTeX* down the road, when the next version comes out. If you're on a *super* tight budget, you could get by with the Oetiker and Flynn electronic books mentioned above (i.e., PDF files).

2.2 Get Up and Running

2.2.1 TEX Users Group TEX Live CD or DVD

We want to get you playing with LATEX as soon as possible. To that end, you need to get a TEX system running on your computer. (Except for the bound books and the Mac OS 9 shareware programs we briefly allude to, everything we mention in this article is free[2], so if you have a computer and a fast internet connection[3], and if you set aside an evening or afternoon, you should have plenty of time to get a TEX implementation installed on your computer.)

If you have a copy of a TEX Users Group TEX Live CD or DVD, you have everything you need to install a TEX system on a Linux/Unix, Mac, or Windows system. You can obtain the TEX Live CD/DVD by becoming a TEX Users Group member, or from some other source such as the two books mentioned above. Please refer to the readme files on the CD/DVD for installation instructions, or check-out the online instructions at:

http://www.tug.org/texlive/doc/texlive-en/live.html

Note that other user groups also provide the TEX Live CD/DVD to their members. For a listing of user groups see: http://www.tug.org/usergroups.html

2.2.2 Linux/Unix

If you are a Linux user, you probably already have TEX in place on your system, and you are good to go. If not, follow your distribution's process for installing packages. For example, we use SuSE Linux and would use their installer YAST to install TEX and its components. Similarly, Red Hat/Fedora users would use that distro's installation mechanism to get TEX ready to run. In either case the process is quite painless. Similarly, if you are using a terminal on a UNIX system, TEX will probably be available already; just touch base with your system administrator for

[2] I do have some limited personal experience with the commercial TEX implementations PC-TEX and V-TEX. If you have questions about those or other commercial implementations of TEX, let me know, and I'll work with you to try to find answers to your questions. —TN

[3] I recall that back in my dial-up days, it would take me more than an hour to download the "small" version 1.2 MiKTEX. I don't dare think how long it would take to download a 2+ version of MiKTEX with a dial-up connection. Now that I have a cable modem, I start to get antsy, whenever downloads take more than several minutes. —TN

details. (Note: The first author is an experienced Linux user, and the second author is a dabbler and wannabe.)

2.2.3 Mac

If you are still using OS 9, you will probably want to install one of the shareware TEX implementations. You can find out more information about these programs at http://www.tug.org/interest.html. If you have OS X, you can use the i-Installer to get TEX on your system. The i-Installer web page is located at:

http://www.rna.nl/tex.html

Be aware of the warning contained in the TEX Live documentation: "The i-Installer distribution uses the teTEX texmf tree with some additions. Due to differences between TEX Live and teTEX you cannot update a TEX Live installation with an i-Installer TEX Program's i-Package."

The TeXShop home page is a good place for Mac OS X users who are interested in TEX to begin their tutelage:

http://www.uoregon.edu/~koch/texshop/texshop.html

2.2.4 Windows

With Microsoft Windows you have two well-established *free* options: TEX Live and MiKTEX. Refer to the section above to learn about the TEX Live CD/DVD. MiKTEX is a popular implementation of TEX, and historically it has been robust and reliable. You can go to the following web page to learn about downloading and installing MiKTEX:

http://www.miktex.org/setup.html

There's a new kid on the block: ProTeXt. It's based on the MiKTEX implementation, but it adds Ghostscript/GSview and a text editor (i.e., TeXnicCenter). Information regarding ProTeXt can be found at:

http://www.tug.org/protext/

3 Project #1

You will come to grips with LaTeX gradually. The second author hates watching football players who can't block and tackle properly (i.e., American football players). In TeX, as in football, there is a core knowledge that every TeX user will need to learn. If they don't, they're bound to have a less agreeable experience. Undoubtedly it's true that we must all crawl before we walk, and walk before we run. This next section will take you step by step through an exercise that is about as simple as possible—the creation of an "Hello World" output.

3.1 Hello World!

For the computer expert, that "Hello World!" implies a minimal program that simply proves you have got everything working. We shall proceed now to generate a "Hello World!" LaTeX program.

First, create a directory where we can play. Let's call it playdir. Change to that directory to keep things simple. Now note that We are going to start you off in LaTeX rather than the more basic TeX. The reason is simple—LaTeX has been designed to make things as easy as possible.

3.1.1 Choose a Text Editor

The next thing you need to do is pick a text editor. It can be as simple as you like since, as will soon become obvious, a LaTeX file contains nothing but the characters you see on your keyboard. The first author happens to use EMACS, because it has some frills that facilitate typing LaTeX stuff, but any text editor will do. The second author uses several editors, including WinEdt, and, in the spirit of full disclosure, we might as will go ahead and state that he uses both MiKTeX and PC-TeX.

3.1.1.1 Windows

In Windows you can use programs like WinEdt or TeXnicCenter, which have been designed specifically for use with TeX. WinEdt is a shareware program, that has a history of regular upgrades, and is supported by a very active mailing list. You can learn about and download WinEdt at:

http://www.winedt.com/

The TEX Users Group offers a discount on the shareware registration fee for WinEdt to its members. For info on these discounts go to http://www.tug.org/winedt/. TeXnicCenter is distributed as open source under the terms of the GNU General Public License (GPL) and is available for free. You can learn about and download TeXnicCenter at:

http://www.toolscenter.org/front_content.php?idcat=26

Both WinEdt and TeXnicCenter are highly recommended by people who use them. You might want to try them both out, and choose your personal preference.

3.1.1.2 Mac

On the Mac, if you're using OS X, you might consider either TeXShop or iTeX-Mac. For more information about TeXShop, see:

http://www.uoregon.edu/~koch/texshop/

For more information about iTeXMac, see:

http://itexmac.sourceforge.net/

3.1.1.3 Linux

If you're running Linux, you probably already have a favorite editor, but, if not, you might consider Kile.

http://kile.sourceforge.net/

3.1.2 Your First LATEX File

We will start by generating a minimal file. Just create a file in your text editor, let's call it hello.ltx, which contains, initially, just the following three lines

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

Save your efforts (i.e., the hello.ltx file) so as not to risk an unfortunate accident. (If you didn't create a playdir earlier, do so now, when you save your file.) Since this is a LaTeX file, we have chosen to give this file an .ltx extension. (You can view or download a copy of the final hello.ltx file we will create online at http://www.tug.org/pracjourn/2005-1/waud/hello.ltx.)

Let's pause and look at what we have. All LaTeX documents have the three lines you typed in your first document. The first line announces that this file is a LaTeX source file (that term "source" simply means it is the starting point for generating the final product, in contrast to other files which may come up, for example, output files). This \documentclass statement has the following general form:

\documentclass[*options*]{*class*}.

The options and class variables are extra bits of information you want to pass on to LaTeX to use when it processes this file. At this point in the game, we aren't going to use any options, and that is why we've omitted the square brackets. In the present case, we have selected a class type (i.e., article). In this statement we are choosing "article" as our class type. The class-type is put in "curly braces" simply to make it easier for LaTeX to pick out, as well as making it less vulnerable to a typo on your part. Alternative class types include book, letter, and report. We need not get into fine differences here; suffice it to illustrate that the article class would not have chapters whereas the book class would—indeed an article class document could be used for a chapter in a book, clever! Note also that, although this may seem an unnecessarily complicated thing to have to add, it makes perfect sense. There is no way LaTeX can read your mind and deduce where you are going. Having to enter details like this is the price you pay for having control. As you learn more details, you can exert finer and finer control.

The other two lines show a recurring theme in LaTeX; that is, the use of *argument* pairs. In this case we have \begin{*argument*} and \end{*argument*} pairs. The *argument* is the thing that is beginning and ending. You have to tell LaTeX *what you want to begin* and then again *what you want to end*—in this case the actual document we are going to generate.

Now, at this point, we really have an empty document inasmuch as there is nothing of our own material in there. We can now create a minimal LaTeX document by typing the line

```
Hello World!
```

between the \begin{document} and its matching \end{document} lines to get

```
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

as our final (for now) document. Again, save it before loosing it.

3.1.3 "LaTeXing" your File

3.1.3.1 Using a Text Editor

At this point, you will have a file hello.ltx in the playdir. Now we need to get TeX to process the file. If you are using an editor designed to work with TeX you should be able to process you file by either clicking on an icon, selecting a menu item, and/or entering a particular keystroke. Those are all great time-savers. Many editors will correctly configure themselves, if they are installed after you have installed TeX on your system. If your editor isn't configured correctly, or you don't know how to "typeset" your file using one of your editor's shortcuts, then please consult the documentation that came with your editor.

3.1.3.2 Using the Command Line

This section is for people who don't have an editor with TeX shortcuts, and people who are curious about how TeX and LaTeX work. If you don't fit into those categories, then skip on to the *Seeing Our End Product* section, which is just ahead.

Okay, let's now see how things work at the command-line level. After all, isn't it nice to know long division, just in case you forget to bring your calculator to the lumberyard? (Mac users are going to get off easy this time—don't they always, you may ask? If you're a Mac user you can either figure out how to use the terminal on your own, or—if you're using TeXShop—you can just click on the "Typeset" and "Preview" menu items at the appropriate time. But make sure that "LaTeX" is selected as your Typeset option. Also, notice that your preview is a PDF file, not a dvi file.)

Linux folks will need to open a terminal window—if you're using either GNOME or KDE, this usually can be accomplished by clicking the terminal icon on the taskbar.

Windows users will need to open a command window. This can be done by clicking on the Windows key, selecting the run command, typing either command or command.com and then pushing the Enter key.

After your terminal or command window is open, switch to the playdir directory. (For example, if you were on the D drive in Windows, you would enter c: to change to the C drive, then enter cd\playdir to change to the playdir directory.)

As soon as you are safely in the playdir directory, type the following at the command prompt:

```
latex hello.ltx
```

You also could have typed:

```
tex &latex hello.ltx
```

This second command would have told TEX to process and format the file using LATEX. The latex command does the same thing, it's just a shortcut. Note that the .ltx extension is not really necessary, since it's one of the default extensions that LATEX looks for (other examples being .tex and .dtx); we just wanted to leave as little to chance here as possible. Also note that the actual LATEX program is simply named latex—all lower case.

After you type in latex hello.ltx and push Enter, LATEX will start to process the hello.ltx file. If you have no typos, LATEX will generate a few lines telling you what it is doing, then LATEX will create a file named hello.dvi. You should now to able to do a directory listing of the playdir directory, and see that there are now several files in the directory—not only hello.ltx, but also hello.dvi (as well as a hello.aux and hello.log, which we shall ignore at this point). (In Windows you can get a listing of the directory by typing dir and pushing the Enter key. You can get a listing of some of the other MS DOS commands by entering help at the command line.)

If you are a Windows' MiKTEX user, and nothing happened after you typed latex hello.ltx and pressed Enter, then most likely your MiKTEX bin directory isn't in your Windows' path. Your Windows' help file can give the information you need to correct the situation: select the Windows key, click on Help, then do a search for environment variable You should also read the MiKTEX user manual.

3.2 Seeing Our End Product

Now we will want to see the end result, but before we admire our handiwork, a bit of background is in order.

When TEX was first developed, computers were much more primitive than today. In particular, printers were few and varied. Therefore TEX was designed not to talk to any particular printer (that would have meant rewriting TEX every time a new printer arrived on the scene). Instead the final step was split into two steps. TEX was designed to put out a "device independent file," with a file extension of .dvi. This file contained the key information necessary to print the file but not anything specific to any particular printer (or screen viewer; in the current context, a viewer is effectively a printer which writes on a computer screen). The final step was achieved with files called "printer drivers." These were programs designed to read .dvi stuff and translate it into the particular language that a given printer "talked." For example, Hewlett Packard designed its LaserJet printers to speak PCL (as in "Printer Control Language"). A driver for the LaserJet would thus be designed to read the code in your dvi file and write PCL code for your printer; that is, send instructions to your printer telling what it has to do to produce the desired output.

Now, since we are interested in a document with only two words, we can save paper and simply view the dvi file; that is, we shall call a viewer to read the dvi file and translate it into commands which will cause the screen to show the final document.

In Linux this again is easy. The screen is Xwindows and there is a driver, xdvi, which will display dvi files there. All we have to do is type

```
xdvi hello.dvi
```

where, again, that file extension, .dvi, is not necessary; after all a program designed to display .dvi files will automatically look for that type of file. But we shall put it in both to be safe and to help remind us where we are in the process.

If you're using Windows and MiKTEX, you can type the following at the command line:

```
yap hello.dvi
```

Yap is MiKTEX's dvi viewer—"yap" stands for "yet another previewer."

You should now have a screen with just the two words "Hello World!" on it. (You may have to scroll around a bit to find the text.)

Now this result tends to be rather disappointing since all you get is a full page with two words on it. But you did get it! This means you have LaTeX up and running and have written your first LaTeX program.

3.3 Review

Let's just review the guts of the whole process.

1. Create an ASCII source file.
2. "latex" the source file.
3. View or print the output file.

That's the basic ritual. Create a file in your text editor. Use a short-cut in your text editor, or go to the command line to run latex on your file. Then view the resulting *.dvi file in your DVI viewer. (Mac TeXShop users will create a PDF, and will view their output in a PDF viewer.) When you get to fancier documents, you may insert stages to create an index or a bibliography, but we'll leave that for future discussions.

4 Tip for Today

We plan to end each column with a "tip" that Doug has personally found helpful. (Mileage varies.) Here is the first sample.

> **Don't be afraid to make mistakes**.

Experience is truly the best teacher.

References

[1] Flynn, Peter. (2003). *Formatting information: A beginner's introduction to typesetting with LaTeX*. *Tugboat*, *23*(2), 115–218.
http://www.tug.org/tex-archive/info/beginlatex/

[2] Kopka, Helmut, and Patrick W. Daly. (2004). *Guide to LaTeX* (4th ed.). Boston, MA: Addison-Wesley Professional.

[3] Mittelbach, Frank, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley. (2004). *The LaTeX Companion* (2nd ed.). Boston, MA: Addison-Wesley Professional.

[4] Oetiker, Tobias, Hubert Partl, Irene Hyna, and Elisabeth Schlegl. *The Not So Short Introduction to LaTeX2e*. Unpublished manuscript. http://www.tug.org/tex-archive/info/lshort/english/lshort.pdf