The PracTEX Journal
TPJ 2005 No 02, 2005-04-15
Rev. 2005-04-17

Travels in TEX Land: Choosing a TEX Environment for Windows

David Walden

The author of this column wanders through world of TEX, as a non-expert, reporting what he observes and learns, which hopefully will be interesting to other non-expert users of TEX.

1 Introduction

This column recounts my experiences looking at and thinking about different ways TEX is set up for users to go through the document-composition to typesetting cycle (input and edit, compile, and view or print). First, I'll describe my own experience randomly trying various TEX environments. I suspect that some other users have had a similar introduction to TEX; and perhaps other users have just used the environment that was available at their workplace or school. Then I'll consider some categories for thinking about options in TEX setups. Last, I'll suggest some follow-on steps.

Since I use Microsoft Windows as my computer operating system, this note focuses on environments that are available for Windows.[1]

2 My random path to choosing a TEX environment

I started using TEX in the late 1990s.[2]

[1] But see my offer in Section 4.

[2] While I started using TEX, I switched from TEX to using LATEX as soon as I discovered LATEX existed. Since both TEX and LATEX are operated in the same way, I'll mostly refer to TEX in this note, since that is the more basic system.

I don't quite remember my first setup for trying TEX. I believe I downloaded it for free from somewhere on the Web, edited with Microsoft's Notepad editor that comes with Windows, and between edits ran TEX in command-line mode. I already had Adobe's Distiller on my machine, so I also used dvips in command-line mode to convert TEX's `dvi` output to PostScript (ps), and then opened a separate window to run Distiller to create a `.pdf` file. I had to have a number of windows open for the various programs in order to move between them for the whole cycle: edit, dvi to ps, ps to pdf via Distiller, and view with Adobe Reader.

I quickly became tired of the feeble Notepad editor and scouted around the Web for something better, found WinEdt,[3] and started using that. From WinEdt I believe I found MiKTEX (also free[4]), but I couldn't (at least didn't) figure out how to configure WinEdt to start TEX, run dvips, run Distiller, and view with Adobe Reader, all by clicking icons at the top of the WinEdt window.

A little while later, I ran into a mathematician who showed me a paper he had written with LATEX. I asked what TEX system he used, and he pointed me toward the commercial PCTEX which he lauded for working "out of the box"—no difficult configuration to do. I bought a copy of PCTEX, it *did* work right out of the box, and I used it for a while.

However, by this time, I had become used to using WinEdt and liked it, so I switched back to trying MiKTEX in combination with WinEdt and Adobe Distiller, which I finally was able to configure to work together by following the excellent instructions by David Arnold of the University of the Redwoods.[5]

At a later date, I briefly tried the commercial Y&Y TEX system (out of business at this point) because an acquaintance had told me it was an excellent TEX system, but again I switched back to the WinEdt–MiKTEX combination with which I was more familiar. Still later, I used the commercial VTEX system[6] to accomplish a particular task that it seemed well suited to doing.[7]

Finally, in preparation for writing this note, I downloaded LATEXEditor[8] to see

[3] `www.winedt.com`

[4] `www.miktex.org`

[5] `http://online.redwoods.cc.ca.us/instruct/darnold/StaffDev/Index.htm`; unfortunately these instructions are out of date for the current versions of WinEdt and MiKTEX.

[6] `www.micropress-inc.com`

[7] Described in my paper "Writing a big book—A first experience with LATEX," *TUGboat*, volume 24, number 2, 2003, pp. 211–215.

[8] `www.ntu.edu.sg/home5/pg03053527/latexeditor/`

how it installed. I also downloaded a 30-day-trial of Scientific Word's commercial setup for LaTeX[9] to get a feeling for how a WYSIWYG version of TeX worked.[10]

3 Some categories for TeX options

Recently, it dawned on me that I wished someone had briefed me on some of the possible ways TeX can be configured to work in different TeX environments before I spent so much time randomly trying various environments. Thus, I set out to make a list of some categories that potential TeX users might consider in trying to decide what was important to them in a TeX environment.

However, before I present my list of categories for distinctions between different TeX environments, I must remind readers that all TeX environments are based on the same underlying TeX program created by Donald Knuth that is freely available to the public.[11] At their core, the different environments are essentially or exactly the same; this is why, to an astonishing degree, all the different TeX environments can be used with TeX documents that have been created within any other environment. Furthermore, all TeX distributions will be using the same LaTeX package (originally created by Leslie Lamport and now maintained by others), and they will have most of the same other packages, styles, auxiliary programs, and so forth.

Now on to my list of categories. The following items are in no particular order.

- How (=to what extent) integrated with an editor.
  The commercial systems (PCTeX, VTeX, etc.) come integrated with their own editor and GUI control panel from which various TeX capabilities can be run and TeX output can be previewed or printed. The non-commercial systems (e.g., MiKTeX) tend to be operated from a command line interface (although they may have a GUI alternative) and require the user to find a separate editor (e.g., WinEdt, Emacs, etc.). In turn, the separate editor must be con-

[9] `www.mackichan.com`

[10] I have also used the combination of PDFLaTeX and Emacs on a Linux system on a few occasions.

[11] See Donald E. Knuth, "The Future of TeX and METAFONT," *TUGboat*, volume 11, number 4, 1990, p. 489 (`http://www.tug.org/TUGboat/Articles/tb11-4/tb30knut.pdf`). Although the program and algorithms are freely available, Knuth puts some restrictions on the use of his program names.

figured to work back and forth with the TEX system (although some sort of configuration wizard may be available, as with WinEdt).

- How oriented the editor is to TEX.
  Of course, the editors in the commercial TEX systems with an integrated editor know about TEX—highlighting TEX commands, having shortcuts to insert command sequences, etc. Some editors have been created especially for TEX, such as LATEXEditor, TEXnicCenter,[12] and Scientific Word.[13] Some general purpose editors (e.g., Emacs and WinEdt) can be configured so they recognize TEX just as they can be configured to recognize HTML, the programming language C, etc.

- Are there other tools that help the user with TEX?
  Are there tools that can (a) shield the user from the TEX commands, (b) help the user learn the TEX commands, or (c) help the user become more efficient? For example, PCTEX has a TEX Helper, and VTEX has an equation editor. If the distribution doesn't come with these types of tools, is it easy to add third party tools?

- How powerful an editor.
  A user who is used to using editors that include a macro capability, ability to search for regular expressions, etc., may want to consider what capabilities are available when the editor comes integrated with the TEX system.

- How out-of-the-box, installed without user intervention, and pre-configured.
  The commercial systems tend install more or less automatically and come with a default configuration that may be satisfactory as is for at least some users. On the non-commercial side, ProTEX t[14] is basically MiKTEX with a PDF document guiding installation. Some stand-alone editors come better set up for one TEX system than for others, and some stand-alone editors try

[12] `www.toolscenter.org`

[13] A complaint some people have against TEX is having to work with the TEX commands. Scientific Word is a TEX system that works substantially in a WYSIWYG mode, although it appears to be primarily set up to create LATEX-based documents. A user still has to click a button to actually compile the underlying LATEX file and to view the result, although much (perhaps most) editing can be done without explicitly looking at LATEX commands.

LyX may provide similar capabilities (`www.lyx.org`).

[14] `http://tug.org/protext`

pretty hard to automatically configure things to work with a TEX system (e.g., WinEdt comes with a MiKTEX default).

- How expensive and how supported.
  The commercial systems have a price list, obviously. Other systems are available for a nominal fee (in the case of shareware), a plea for contributions, or with a society membership (e.g., TEX Live comes with membership in the TEX Users Group[15]). For many users, it is worth paying the commercial price to have a system that works "out of the box," has a degree of formal technical support or warranty, and has special features a particular vendor has provided. All TEX users can get general TEX support from the `comp.text.tex` newsgroup, `texhax@tug.org`, etc., lists. Some TEX systems have their own discussion groups (e.g., `www.miktex.org/fora.html`, `protext@tug.org`, `tex-live@tug.org`, `yandytex@lists.ucc.ie`, and `mactex@tug.org`).

- How stable is the system.
  There are several ways to consider stability. First, whether the system actually has a substantial number of bugs, although any of the well-established systems out of their infancy should not have much problem in this area. A variation on this first issue is how fast problems are fixed by the vendor or provider when a problem is demonstrated. Second, many systems have two versions—a stable system, and a system with new capabilities or bug fixes that is available for beta-test by users who want the new capabilities more than they worry about hitting an occasional bug. A third aspect of stability is whether the vendor or provider is pushing upgrades with new capabilities with fair frequency, or trying to minimize changes over time which minimizes the needs for users to bother with upgrades. Fourth, some systems have a substantial research component (e.g., the work by the developers of the Omega version of TEX).

- Features
  Most systems claiming to be TEX provide (as least optionally) a large set of TEX-related capabilities, including a variety of TEX-based macro packages

[15] The TEX Live system is produced as a joint effort of all of the major TEX user groups and is also available free as a download.

(e.g., LaTeX, AMSTeX, and ConTeXt), a large collection of style packages, modifications to basic TeX to provide additional capabilities (e.g., PDFTeX and e-TeX), and many helpful utilities.[16] Beyond the large set of TeX capabilities that come with all TeX systems, some vendors and providers provide additional capabilities or options which a potential user many want to know about. Also, while every TeX system provides a significant set of fonts, one way that TeX vendors distinguish themselves is by providing additional sets of desirable fonts that come with their basic system or as options. Another issue is how well the system is set up to easily handle different languages (e.g., hyphenation patterns), if the user needs different languages. Finally, there is the question of what limitations a system from a particular vendor or supplier may have.

- Update and configuration capabilities.
  The TeX world is a big world and options and updates can come from many places. Different vendors and suppliers of TeX have different approaches and levels of support for getting options and updates including in some cases configuration or update wizards.

- What do my friends and colleagues use—people I can ask questions of.
  While there is lots of publically available support for TeX, it is very nice to be able to ask the person across the hall or with whom you are already in communication for help.

4 Next steps

I am not interested in doing comparative evaluations of the various TeX systems. However, I do think a quasi-taxonomy for noting similarities and differences among systems can be useful to relatively new users of TeX. Therefore, I solicit your feedback on the discussion in Section 3, suggestions for different or better categories, improvements for my descriptions of the categories, and so forth.

At this point, this note only considers configurations of TeX that run under MS Windows (the operating system I use), that is, LyX, MiKTeX, PCTeX, ProTeXt,

[16] One area someone considering acquiring a TeX system should be clear on is what set of capabilities the system has to produce dvi, PostScript, PDF, and HTML outputs.

Scientific Word, TEX Live, TrueTEX, VTEX, ... I will be happy to expand my list to include versions of TEX that run on other operating systems, if readers give me information about these.

Also, if vendors, suppliers, or users of various TEX systems tell me what they think the distinctive aspects of their TEX environments are, I will also try to put these into some sort of chart.

I will then provide a revised version of this paper that includes new information I have received.

Acknowledgments

Karl Berry, Tim Null, Steve Peter, and Christina Thiele looked at drafts of this note. Two reviewers also made specific and detailed comments. I am grateful to all of these people for taking the time to help me as well as for their specific suggestions for improvement, some of which I copied verbatim.

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. More history is at `www.walden-family.com/dave`.