

Travels in \TeX Land: Word2 \TeX redux, \TeX 2Word, plain \TeX with Eplain, and playing with “thought breaks”

David Walden

Abstract

In this column in each issue I muse on my wanderings around the \TeX world. In the last issue, I described my trial of Word2 \TeX . In this issue, I first describe a little more of my experience with Word2 \TeX , second describe a trial of \TeX 2Word, third describe my efforts to try \TeX itself (not \LaTeX) for the first time, and fourth look at several ways to typographically show a change of subject.

1. Word2 \TeX redux

In my last column (this journal, 2005, number 3), I mentioned my trial use of Word2 \TeX from Chikrii Softlab (www.chikrii.com). Shortly afterwards, a reader sent the following message that I’ll answer below:

The Word2 \TeX section left me hanging... did the full version convert math in the papers? Did it convert all the papers smoothly? What is the price and is it reasonable? Given that many Word authors use Equation Editor, does it convert those glyphs? Maybe it would be worth taking out this section and having a separate review of Word2 \TeX in another issue...

I appreciate what the user is asking for, but I am probably not the right person to provide a thorough review of any piece of software: I tend to use just enough of any software package to get my job done and never look into most capabilities, and I am not very fussy about a piece of software as long as it helps me do my job even if I still have to do lots of manual work. Also, I mostly don’t encounter equations in my writing and editing, and I know nothing about Equation Editor and so can’t answer about its glyphs. You can read what the vendor says about its product at <http://www.chikrii.com/dl/word2tex/word2tex.pdf>. The manual mentions Equation Editor, Mathtype, and regular text equations; and according to the company the product “converts all equations created by all

existing versions of Equation Editor and MathType.” You can see equation conversion in the trial version, although only for the first seven equations in the Word document.

However, I *can* say a little more about my experience with Word2 $\text{T}_{\text{E}}\text{X}$. The price is listed as \$99 (with an academic list price of \$45). I bought a copy for \$85 from PCT EX (www.pctex.com). For me, \$85 was a reasonable price because it was less than hundreds of dollars and because the program has saved me lots of keystrokes in manually converting Word documents to $\text{L}\text{A}\text{T}_{\text{E}}\text{X}$. Word2 $\text{T}_{\text{E}}\text{X}$ works well for me in the sense that it converts the whole Word document into a valid $\text{L}\text{A}\text{T}_{\text{E}}\text{X}$ document, and does this fast. In the cases I have used it, Word2 $\text{T}_{\text{E}}\text{X}$ has not made an exact copy of my Word document, and its decisions of the best $\text{L}\text{A}\text{T}_{\text{E}}\text{X}$ command(s) for the conversion of a specific element from Word often need to be changed to something else once I am working on the resulting $\text{L}\text{A}\text{T}_{\text{E}}\text{X}$ document. However, I think it is to be expected that one will have to fine tune the resulting $\text{L}\text{A}\text{T}_{\text{E}}\text{X}$ to get it to produce exactly the document you want. I don’t see how any program could know enough to convert an arbitrary Word document into exactly the $\text{L}\text{A}\text{T}_{\text{E}}\text{X}$ the user wants. The Word2 $\text{T}_{\text{E}}\text{X}$ manual mentioned above appears to give some tips for writing Word documents in a way that improves Word2 $\text{T}_{\text{E}}\text{X}$ ’s ability to make the most sensible conversion (but I use the system without benefit of having read the manual).

Although I bought and downloaded Word2 $\text{T}_{\text{E}}\text{X}$ from PCT EX , I had to interact with the Chikrii web site to get the needed code words to make the downloaded software operational. This step was straightforward enough. This “registration” process enables Word2 $\text{T}_{\text{E}}\text{X}$ for use only on the specific computer that is being used for the interaction with the Chikrii web site (site licenses are also available, according to Chikrii’s web site).

Although Chikrii Softlab has recently released Word2 $\text{T}_{\text{E}}\text{X}$ 4.0, my experience was with Word2 $\text{T}_{\text{E}}\text{X}$ 3.0.

2. $\text{T}_{\text{E}}\text{X}2\text{Word}$

Also after the appearance of my last column with its notes of Word2 $\text{T}_{\text{E}}\text{X}$, the VP of Marketing for Chikrii Softlab suggested I should try $\text{T}_{\text{E}}\text{X}2\text{Word}$ for this column. I said that I would try it and download the trial version (good for 30 days) of $\text{T}_{\text{E}}\text{X}2\text{Word}$ 2.0 (version 3.0 is scheduled for release soon, according to the company). The trial version has limitations on how many equations (i.e., something embedded in a pair of dollar signs) it will process and also a limitation on the number of figures. But I only had two figures, so that limitation didn’t get in my way.

The process of downloading and registering $\text{T}_{\text{E}}\text{X}2\text{Word}$ was the same as

it was for Word2T_EX, as described in my prior column (TPJ 2004-3). The price is also the same—\$99 list. To convert a T_EX document to Word, you start Word, click Open file, set the file type in the open-file dialog box to .tex, select the file you want to convert from T_EX, and click OK. New content appears in the Word editing window that is the conversion of the .tex file.

The document I converted to Word as my trial of T_EX2Word was about 10,000 words long and 22 PDF pages long when compiled by PDFL^AT_EX. It contained no real equations (but a number of instances of paired dollar-signs where I was trying to superscript a comma, e.g., $\$^{\{, \}}$) and only two figures. This conversion was a real project, not a toy. It is one of those instances where a paper was written in L^AT_EX but the journal publisher insisted on it being submitted as a Word file. The conversion from T_EX to Word takes considerably more seconds than a conversion of the same size document from Word to T_EX. However, this was not a problem for a 22-page document.

Automating going from T_EX to Word seems harder than automating going from Word to T_EX, at least for the aspects of T_EX that I use versus the aspects of Word that I use. For instance, and perhaps not surprisingly, T_EX2Word doesn't know how to handle new definitions in a T_EX file. On the other hand, it does a good job of conversion at the character level.

As with Word2T_EX, a good bit of cleanup was required after my T_EX2Word conversion; but, as with Word2T_EX, I can't imagine a program that could correctly guess what the perfect conversion would be.

All in all, using T_EX2Word saves a good bit of time over much more manual methods.

3. Trying plain T_EX

I have had *The T_EXbook* since before I discovered that I would mainly use L^AT_EX and bought a book on that, but I don't remember ever writing a significant document using T_EX. I also have long had copies of *T_EX for the Impatient* by Paul Abrahams et al. and *T_EX by Topic* by Victor Eijkhout that I downloaded from the Internet and printed. Then, a while back, I got tired of how much space on my bookshelf was taken by the downloaded copies of the Abrahams and Eijkhout books (eight-and-one-half-by-eleven, printed-on-one-side pages in loose leaf notebooks); and I bought used copies of the Abrahams and Eijkhout books.¹

Having bought the used books stimulated me to think about T_EX itself

¹ The Abrahams book is readily available at a relatively inexpensive price; the Eijkhout book is hard to find and is rather pricey.

again, and I eventually decided to give it a try.² Then, on September 21, 2005, a message came from Oleg Katsitadze to the `texhax` list announcing version 3.0 of Eplain (<http://tug.org/eplain/>).³ So, I downloaded the `eplain-3.0.tar.gz` file, found some programs that knew how to deal with `.gz` and `.tar` formats, installed the whole `eplain-3.0` directory in my `\localtexmf\tex` directory, and tried writing this column in plain \TeX with Eplain augmentations. Here is a bit about my experience.⁴

First, I began preparing this column doing as little study as I could of \TeX or Eplain. You can see these first steps done with minimal \TeX in the file `test-1.tex` on the HTML page for this paper. I defined a little placeholder `\url` command in lieu of looking up what `\url` capability was available in Eplain. Otherwise, I don't think I used anything in \TeX that I didn't already know from using the \TeX core of \LaTeX .

As I continued to write the column, I added some \TeX commands to make this paper look more like a paper done with the \LaTeX style for this journal. The source file of this is shown in the file `test-2.tex`. I am sure I did not use good \TeX practice. In fact, I did the minimal looking up of stuff in my books on \TeX to find a handful of \TeX commands that helped me approximate the look of papers using the normal \LaTeX `pracjourn` class, for example:

```
\narrower
\font\smallrm = pplr7t
\font\tenit = pplri7t at 12pt
\baselineskip = 14pt
\newcount\sectionnumber
\advance\sectionnumber by 1
\beginsection
```

I have never known anything about fonts in \TeX or \LaTeX but reading a tiny bit about the `\font` commands in \TeX gave me the first clue. Of

² By trying " \TeX itself," I mean clicking the `PDF \TeX` rather than the `PDF \LaTeX` button on the control panel of my \TeX -oriented text editor.

³ Karl Berry originally developed Eplain. Oleg Katsitadze currently maintains and improves it.

⁴ I have been asked, "*Why* try plain \TeX ? What does it give that \LaTeX does not?" This column reports my wanderings in the world of \TeX , and my wanderings are not necessarily motivated by logic. As I explained in the main text, this look at plain \TeX was stimulated by wishing I did not have two thick notebooks on my book shelf. However, I can give two logical reasons for knowing more about and using plain \TeX : 1) like most everyone, I use some plain \TeX commands from within \LaTeX , and knowing more about \TeX may prove useful in using \LaTeX ; 2) sometimes I develop a document where I want to start with more of a "blank page" than the basic \LaTeX classes provide, and it may be easier to start from plain \TeX than to try to undo the assumptions of a \LaTeX class.

course, I have seen explanations of this many times over the years in various L^AT_EX books, but this was the first time I paid enough attention to begin to understand it. I now understand that among all the other font stuff that may be in my main T_EX file structure, there are lots of files with a `.tfm` extension. These files include sufficient descriptions of the dimensions of characters in a font for T_EX to lay out a page without knowing what the characters actually look like.

I did a file name search over my `\texmf` directory and file structure that resulted in a long list of `.tfm` files. Looking at all these file names gave me the motivation to try again to read a description of Karl Berry's naming system (based on 8.3 type DOS file names⁵) for `.tfm` files. Thus, `pp1r7t.tfm`, for instance, is a file containing the size descriptions for Adobe's (the first p) Palatino (the pl) regular weight (r—e.g., not bold) font using the original T_EX encoding (7t).⁶ The original Computer Modern fonts from Knuth do not follow Karl Berry's naming system and continue to use their original names, e.g., the `cmr10.tfm` file contains the sizes for the Computer Modern Roman 10 point font. Also, there appear to be many `.tfm` files that do not adhere to Berry's 8.3 naming system, e.g., `ecobx2488.tfm`—I guess some of the 8.3 limitations of DOS are becoming a thing of the past.⁷

Try looking yourself at all the `.tfm` files in your T_EX distribution. It is educational. There are 16 subdirectories in my `\texmf\fonts\tfm` directory, e.g., subdirectories for `adobe`, `ams`, and `yandy` fonts. By far the biggest subdirectory of `\texmf\fonts\tfm` is `public` which itself has well over 100 subdirectories of different sets of fonts.

Finally, as I finished writing the column, I skimmed the Eplain documentation a bit and added some of its capabilities as shown in the file `test-3.tex` where I used most of the following Eplain commands:

```
\numberedfootnote ... \everyfootnote
\verbatim ... \endverbatim
\verbatimescapechar
\unorderedlist ... \endunorderedlist
\enablehyperlinks
\beginpackages
  \usepackage{url} %required downloading url.sty
\endpackages
```

⁵ Karl Berry, *Fontname: Filenames for TeX fonts*, <http://lulu.com>, public domain.

⁶ 8t, for instance, indicates the Cork 256-character encoding, but for my purposes the original T_EX encoding is all I will probably ever need.

⁷ Steve Peter told me that the widely used `fontinst` tool continues to require the 8.3 naming scheme for its input files.

Notice the hyperlinks with URLs and footnotes in this file. Oleg Katsitadze says, “Starting with version 3.0, Eplain natively supports hyperlinks with all its cross-referencing macros, much like L^AT_EX with the ‘hyperref’ package.”⁸

I did have one problem using Eplain. The file for this column would not compile with PDF_TE_X. I could compile it by creating a dvi file with T_EX and then converting the dvi file to a PDF file. I reported this problem to Oleg Katsitadze, who was very responsive about investigating the problem and letting me know its disposition.⁹

I have several thoughts from my brief and superficial experiment with T_EX and Eplain:

- Using minimal capabilities of T_EX and Eplain was not very hard.
- Obviously it would have been a lot more work to find and use capabilities in T_EX equivalent to those I routinely use in L^AT_EX.
- I will probably do more simple documents in T_EX itself, as it will help me learn more about T_EX which I would never learn just by using L^AT_EX or by reading T_EX documentation since I’m disinclined to study documentation in any rigorous way.
- I can imagine that if I gather a nice set of T_EX macros together over time, I might prefer T_EX for some uses, for instance, for little one-off projects where sometimes L^AT_EX’s standard classes (e.g., `article`, `book`, ...) just get in the way. On the other hand, it really is nice that one can give a single command such as `\section` in L^AT_EX that will change the font size, do appropriate things with vertical line spacing, increment the section number, and so forth.¹⁰
- However, probably there is lots more being done in L^AT_EX than T_EX these days and, thus, L^AT_EX probably remains the high leverage place to work.

⁸ The current default in Eplain is to put boxes around hyperlinks as well as using a colored font for hyperlinks. According to Oleg Katsitadze, such boxing is consistent with the PDF default for hyperlinks. To remove the boxes, he advised me to put the command `\hlopts{bwidth=0}` after `\enablehyperlinks`.

⁹ According to Oleg, the problem is that in the distribution of Mik_TE_X that I use (from TUG’s Pro_TE_Xt CD), `supp-mis.tex` uses `\fmtname` to detect plain TeX (plain T_EX defines `\fmtname` to be the word “plain”), and since Eplain changes that to “eplain,” `supp-mis.tex` thinks it is dealing with L^AT_EX and, thus, fails to compile my T_EX-Eplain file. To fix this, one can define `\fmtname` back to “plain” after `\input eplain`. This problem arose because Eplain 3.0 allows some L^AT_EX packages to be used (a *neat* feature), and one of the L^AT_EX packages I used the file `supp-mis.tex`. Since `supp-mis.tex`, in turn, is associated with Con_TE_Xt, resolving this problem apparently will involve discussion between the maintainers of Eplain, Con_TE_Xt, and Mik_TE_X.

¹⁰ I created such a macro for use with T_EX, copying and then modifying (but not fully understanding) the definition of `\beginsection` from page 255 of *The T_EXbook*. See my definition of `\section` in the file `text-5.tex`.

- Nonetheless, I look forward to trying plain \TeX and Eplain some more.

4. Thought breaks

In my writing of papers and books, I sometimes use what I call “thought breaks”—an extra vertical space and perhaps a few letters of font change at the beginning of a paragraph that begins a new topic but not a topic so different as to deserve its own chapter or section title.¹¹ I have not found an archive of various ways of doing thought breaks in \TeX , so I have been developing my own little library. For instance, see the way a thought break is indicated between this and the next paragraph.

I discussed the macro for this approach to a thought break in some detail in *TUGboat*.¹² It is defined as follows:¹³

```
\def\thoughtbreak#1{\bigskip\noindent{\Large #1}}
```

Since the above definition uses `\Large`, it only works when I am using \LaTeX , but this is most of the time.

I also frequently have used a simpler version of this:

```
\def\thoughtbreak{\bigskip\noindent{}}
```

This version just put in an extra vertical space and inhibits indentation.

The advantage of including something graphical as part of a thought break is that a thought break is still easily recognized at a page break. Karl Berry suggests inhibiting page breaks at thought breaks if thought breaks are only indicated with extra vertical space. In this case, he would rewrite the definition in the previous sentence to be something like:¹⁴

```
\def\thoughtbreak{\nobreak\bigskip\nobreak%
  \noindent{}}
```

At present I am in the final stages of drafting a book, and I have been considering fancier typographic devices for thought breaks. For instance, I looked into the `lettrine` style, which works as follows in \LaTeX :

```
\usepackage{lettrine}
\def\thoughtbreak#1#2{\bigskip\noindent
  \lettrine{#1}{#2}}
\thoughtbreak{T}{his is} the start of a paragraph
```

¹¹ I don’t know what the world of typography calls these; if anyone knows a generic name for them, please let me know.

¹² Vol. 25, No. 2, pp. 199–201.

¹³ I use `\def` instead of `\newcommand` in these examples because a) I am writing this column using plain \TeX , and b) because in some cases I will use \TeX ’s capability for having macro parameters be delimited with other characters than braces.

¹⁴ Karl says, “The second `\nobreak` attempts to inhibit a page break at the `\baseskip` glue before the first line of the following paragraph. Undoubtedly there are cases this doesn’t handle, but this gives an idea of what might be done.”

In this case the letter “T” becomes a dropped capital two lines high, and the letters “his is” are in small caps. I also use another version of this when I do not want to include more than the first word in the dropped-cap-small-caps group of letters:

```
\def\thoughtbreak#1#2 {\bigskip\noindent
  \lettrine{#1}{#2}\ }
\thoughtbreak This is the start of a paragraph
```

In my book, I may end up using a small horizontal line to delimit the thought breaks, as above.¹⁵

```
\def\thoughtbreak{\vskip2pt
  \centerline{${\vrule width2cm height .5pt}$}
  \vskip2pt\noindent}
```

A slight variation is to use * * * or some other graphic instead of a rule.¹⁶

William Adams notes that there is a lot of flexibility in how the break indications may be placed and points to his use of the “Bordas” style of indicating paragraphs (see <http://members.aol.com/willadams/portfolio/typography/theriver.pdf>).

In a September 11, 2003, posting to `comp.text.tex`, John Culleton described what he called a “centered versal instead of a dropped cap versal.” Like the example from William Adams, Culleton’s method lets the thought break be in the middle of a line. **T**HIS is an example of using Colleton’s method. It is implemented with a macro such as the following:

```
\def\thoughtbreak#1#2{\noindent\hglue .4
  \hsize{\tfd#1}{\sc#2}\ }
```

In the above definition, `\tfd` is a font command for a font like `\Large roman` in \LaTeX , and `\sc` is small caps.

THE FINAL THING I wanted to do in my trial of \TeX (see Section 3) was to try including a graphic. So I have included the following image where we see

¹⁵ For a discussion of how to make a “swelled rule” for this purpose, see Steve Peter’s paper elsewhere in this issue.

¹⁶ The definition I would like someone to provide me with is the one where less space is left above the rule (or asterisks or whatever) when the right end of the line above is to the left of the left end of the rule.

a thought break that involves a vertical space with a stylish symbol and all caps on the first phrase of the following paragraph.¹⁷

“You may already know the NANCY DREW books.” This latter was a rich display of false modesty if ever there was one, for by the end of 1931, there were very few people, especially people involved in the juvenile book world, who had not heard of Nancy Drew.



THE OPENING PAGES of Nancy Drew Mystery Story number seven, *The Clue in the Diary*, published in 1932, noted that, in addition to all of the teen sleuth’s other stellar qualities, “in any crowd, she unconsciously assumed leadership.” Even when her peer group

Acknowledgments

Karl Berry and Oleg Katsitadze pointed out mistakes and misconceptions and provided other tips. Philip Taylor caught several typos and advised me on a philosophical issue. Steve Peter did a proofreading pass, caught several errors, made other useful suggestions, and showed me some commands to include an image. Will Robertson also did a proof reading pass and pointed out a number of issues. Although I asked Gregory Kapsias and Kirill A. Chikrii of Chikrii Softlab to review sections 1 and 2 for accuracy, my assessments are my own. William Adams helped with a question I had about typography. The two anonymous reviewers provided many helpful comments and caught a good number of typos.

¹⁷ From Melanie Rehak’s *Girl Sleuth: Nancy Drew and the Women Who Created Her*, Harcourt, 2005, p. 154.