The PracTEX Journal
2009
Comment on this article

Travels in TEX Land: A first attempt to use XETEX (with Windows XP)

David Walden

Abstract XETEX has excited the TeX community over the past few years because it allows fonts on the operating system to be used without them being configured to be TeX fonts. Another cause for excitement is direct input of Unicode. XETEX is included with the September 2008 TeX Collection DVD, both in the TeX Live and ProTEXt distributions. So, I decided to try XETEX on my Windows XP machine.

1 Installing the latest edition of ProTEXt

I'm always afraid to change anything about a working application, especially when I regularly need the application in the course of my work. Since I constantly use TeX on my desktop machine, I decided to try XETEX on my laptop machine.

I put the DVD in my laptop's DVD drive and clicked the ProTEXt option, because I have been using ProTEXt (which is based on MiKTEX) for several years and didn't want to learn anything new. The ProTEXtinstallation is guided by an Adobe Reader script. The first step the script says is required is to uninstall the current version of MiKTEX, which I did. Next it says to install the new version of MiKTEX, which I also did (the complete rather than typical option). This was scary because it gave me four error messages:

```
MiKTeX207-core.dll is missing
MiKTeX207-core-PS.dll is missing
packagemanager.dll is missing
packagemanager-PS.dll is missing
```

However, things seemed to work in any case. (A later correspondence with Thomas Feuerstack, ProTEXt creator, revealed that his installation log file also shows these

files as missing but they are actually in the file `C:\Program Files\miktex\bin`; also Thomas noticed in a log from updating of the system (`Start -> MiKTeX 2.7 -> Update`), these files are flagged as "unregistered." Thomas told me that he thought it was safe to ignore the error messages.)

Next the ProTEXt script said I was required to install TEXnicCenter, which I didn't want to do since I use WinEdt. I kept reading the instructions (beyond the procedure for installing TEXnicCenter) and found it said there that WinEdt is also an option. I guess by "required" the script meant that you don't have a complete TeX installation without both TeX and a text editor, not that TEXnicCenter is the required editor. I think this could be made more clear.

The rest of the ProTEXt instructions are about installing Ghostscript, but I don't use that so I didn't install it (I use Adobe Acrobat and the Acrobat Reader).

2 Trying XƎTEX

I looked around for some instructions on how to have a LaTeX file tell XƎTEX to use a particular font and found some sketchy information at the the XƎTEX website (http://www.tug.org/xetex/ and XƎTEX wikipedia article (http://en.wikipedia.org/wiki/XeTeX). (Since I originally drafted this note, a list of many sources XƎTEX information has been compiled: http://tug.org/xetex/.) Those sources said to include commands like the following in the LaTeX file:

```
\usepackage{fontspec}
\usepackage{xunicode}
\usepackage{xltxtra}
\setmainfont[Mapping=tex-text]{Lucida Sans Unicode}
```

But when I gave the command

```
xelatex myfilename.tex
```

at the command line, it didn't work—it was as if the font command was never seen and the file compiled using the default Computer Modern font.

I read some more and found that apparently XƎTEX is supposed to find where Windows keeps its fonts by looking at some file among those in the "Documents and Settings" directory. I looked there, and didn't find anything that had been installed by MiKTEX (given, perhaps, how I installed it).

I looked around some more and found *The XƎTEX Companion*, February 2, 2009, edition (cern.ch/XML/lgc2/xetexmain.pdf), and on page 33 found some "Supplementary commands introduced by XeTeX."[1] The *Companion* gave an example of the command

```
\font\myname="[/mydir/myfontfile/]"
```

I tried this with

```
\font\MYfont="[c:\mydirectory\fonts\ANTQUABI]"
```

but this didn't work because TeX thought the backslashes in the command introduced non-existent TeX commands. So, I moved the fonts directory into the directory of the TeX file and gave the command

```
\font\MYfont="[fonts/ANTQUABI]"
```

and this did work.

Next I tried another font:

```
\font\MYfont="[fonts/VIVALDII]"
```

but xelatex gave an error message and failed to produce a PDF file. After trying various miscellaneous things for a while, I deleted the .log, .aux, and .pdf files from the original successful compilation of my file, and then it did work to recompile the file with the new font specification. Eventually it occurred to me that it was sufficient just to close the prior instance of the compiled PDF in Acrobat Reader to enable a recompilation without an error from xelatex.

I tried two more fonts with the file test.tex containing the following:

```
\documentclass{letter}
\begin{document}

\def\line{ABCDEFGHIJKLMNOPQRSTUVWXYZ\\
          abcdefghijklmnopqrstuvwxyz1234567890}

\font\myfonta="[fonts/VLADIMIR]" \myfonta \line\\
```

[1] I also found Will Robertson's document on his fontspec package — more about that (better approach) later.

```
\font\myfontb="[fonts/ANTQUABI]" \myfontb \line\\
\font\myfontc="[fonts/CALIFB]" \myfontc \line\\
\font\myfontd="[fonts/arial]" \myfontd \line\\
\font\myfonte="[fonts/pala]" \myfonte \line
\end{document}
```

which resulted in the output in the file test.pdf.

Before going on, I decided it was time to page a bit more through the *The XƎTEX Companion*. Still, after quite a few hours of reading and trial and error, it did not work to define a font usable with XƎTEX using a command such as

```
\font\MYfont="Book Antiqua" \MYfont
```

So, before going to bed for the evening I sent a message to `protext@tug.org mailing list` and joined the MiKTEX user mailing list (http://www.miktex.org/list). In the morning I had responses from both Thomas Feuerstack, ProTEXt's creator and maintainer, and from Ulrike Fischer (as well as others on the MiKTEX user mailing list).

Ulrike made two useful points: (1) make sure Windows is showing hidden files so you can see if the MiKTEX font configuration files are in the Documents and Settings hierarchy, and (2) give the command

```
fc-cache -f
```

to renew MiKTEX's font cache. Things worked a lot better, once I could see

```
\fontconfig\config\localfonts.conf
```

and

```
\fontconfig\config\localfonts2.conf
```

in the Documents and Setting hierarchy; I had augmented the latter with the path `C:\WINDOWS\Fonts` (where Windows XP keeps its fonts); and I refreshed the font cache. I then was able to experiment with accessing many fonts using the same command as above (that failed before refreshing the cache):

```
\font\MYfont="Book Antiqua" \MYfont
```

where `Book Antiqua` was one of the files in `c:\Windows\Font`.

3 Expanding my experiment with XeTeX

As a next step, I decided to try to use what I knew so far to compile this column. First I just included a single XeTeX font command, like one of those shown above:

```
\font\myfonte="[fonts/pala]"
\myfonte
```

This column seemed to compile for a bit, but then failed when trying to handle the footnote. Apparently the pracjourn style uses a different font for footnotes than it uses for the main text, and it was undefined. I removed the footnote and tried recompiling, but then it failed when it came to a `\url` command. Next I looked at this column compiled by pdfLaTeX instead of XeTeX (with the XeTeX font command removed), and then I looked at the fonts that the resulting PDF included.

The files were CMSS10, CMTT12, and URWPalladioL-Bold, -Ital and -Roma. It was time to read about Will Robertson's fontspec package (mirror.ctan.org/macros/xetex/latex/fontspec/), of which he notes,[2] "In XeTeX you probably don't want to use `\font` unless you know what you're doing, whereas `\setmainfont` and so on allow you to use italic/bold fonts as well."

I (sort of) followed the instructions on page 4 of Will's document and inserted the following commands in a test file:

```
\usepackage{fontspec}
\setmainfont{Helvetica}    %the odd choices of fonts here are
\setmonofont{Playbill}     %  so I can easily see that something
\setsansfont{Minion Pro}   %  is changed from the TeX defaults
```

and then generated a line of output with the default font, the `\ttfamily` command, and the `\sffamily` command.

I went back to the LaTeX file for this column, and inserted the following in the file:

```
\usepackage{fontspec}
\usepackage{xunicode}
\setmainfont{Bookman Old Style}
```

[2] April 26, 2009, email.

```
\setmonofont{Courier}
\setsansfont{Helvetica}
```

It compiled and produced a PDF output file, although there was a warning that LaTeX doesn't handle micro-typesetting and I should use pdfLaTeX instead. Since I hadn't specified micro-typesetting, I looked in the `pracjourn` class and found it specified micro-typesetting. I created my own version of the `pracjourn` class (`my-pracjourn.cls`) with the command to use microtype package commented out.

The log file also included the following font warning

```
Font shape 'TS1/BookmanOldStyle{0}/bx/n' undefined
using 'TS1/cmr/m/n' instead
for symbol 'textcopyright' on input line 68
```

and XeTeX didn't produce a PDF output file. Deleting the `\copyright` command from the file eliminated this message. I didn't know how to get XeTeX with the fontspec package to know about `\copyright`, so I moved on (but see section 5).

At this point using the fontspec commands for setting the main, mono and sans fonts, I had access to everything allowed by LaTeX NFSS font commands for family, series, and shape.

As a next step (for now), I tried a fontspec command for including text in an arbitrary font beyond the main, mono and sans fonts:

```
\newfontface\TEST{Arial Narrow Bold Italic}
{\TEST  This is a test}
```

It worked.

I also tried the command for using an arbitrary font family:

```
\newfontfamily\TESTb{Century Gothic}
{
\TESTb  This is a test\par
\textbf{This is a test}\par
\textit{This is a test}
}
```

This also worked.

Will Robertson emphasizes the following:[3]

> `\font="[...]"` loads an "external font," selected with a (path+)filename (such as `"fonts/pala"`), whereas `\font="abc"` or `\setmainfont{abc}` use the "font name" such as "Linotype Palatino." Note: you can also use something like
>
> ```
> \setmainfont[ExternalLocation] {fonts/pala}
> ```
>
> to load an external font through fontspec.

There is much more to learn about XeTeX and the fontspec package, but for now I have enough to start on a significant project.

4 Interim conclusion

XeTeX is pretty neat. It's nice to have access to the fonts on the operating system which aren't necessarily available in TeX. However, getting things configured to work with a new version of MiKTeX and using XeTeX for the first time was a many hour job. As always, for me, installation and configuration is far and away scariest and often the most difficult part of using a new capability. The documentation that was available for this set of experiments is impressive in its volume. However, as so often is the case, I seem to struggle with various problems as I get enough insight into the capabilities of the system to be able to understand what the documentation is telling me.

Another problem was that in some cases when XeTeX didn't find the needed font I specified (for a reason I didn't understand, even though I the font I specified seemed to be on the machine), it hung up and I had to close the command line window to get back control. I can live with this for now.

A problem for me personally with using XeTeX is that I don't have a good enough eye for fonts to be able to see when XeTeX is using the font I specify. To know what font is being used, I have to load the resulting PDF file into Acrobat Reader, give the Properties command, and click the Fonts tab to see which fonts are embedded.

[3] Ibid.

5 Afterword

Before I put this column to bed, I decided to look into the issue of getting the error message

```
Font shape 'TS1/BookmanOldStyle{0}/bx/n' undefined
using 'TS1/cmr/m/n' instead
for symbol 'textcopyright' on input line 68
```

when I included `\copyright` in the file being compiled with the pracjourn class. I looked at the file `pracjourn.cls` and it does not define the `textcopyright` symbol. I looked in the file `latex.ltx`, and it defined `textcopyright` in terms of `\textcircled{}` which I didn't see in the index of *The TEXbook*, so perhaps it is defined somewhere else in LaTeX. Instead of searching farther, I took a look at *The TEXbook* definition of `\copyright` (on page 356). This didn't involve the `textcopyright` symbol, so I inserted the definition from the book in my file,

```
\def\CR{{\ooalign
   {\hfil\raise.07ex\hbox{c}\hfil\crcr\mathhexbox20D}}}
```

and used `\CR` rather than `\copyright`. This worked, and to my eye the result was a satisfactory work-around.

Acknowledgments

I have already mentioned that people responded to my queries to the MiKTeX and ProTeXt lists; I greatly appreciate their help. Karl Berry also helped with one of the problems I faced and spotted a number of typos. Will Robertson spotted some typos and suggested several other useful notes for me to include. I also appreciate the help of the participants in the XeTeX discussion group: http://tug.org/mailman/listinfo/xetex

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. He holds an undergraduate math degree and completed a graduate school sequence of courses in computer science. More history is at www.walden-family.com/dave.