Extending primitive coverage across engines

Joseph Wright

1 The pdfTeX situation ...

In recent years, development of pdfTeX has intentionally been limited, with the v1.40 branch now being around for over 10 years. However, in the past there were plans for a v1.50 branch, and some code was written. One primitive that was fully coded-up at that time was `\expanded`. The idea of this is pretty simple: it carries out full expansion like `\message` (and *almost* like `\edef`), but is itself expandable. This highly useful idea made it into LuaTeX (which was initially based on the pdfTeX development code), but until recently wasn't in released pdfTeX itself.

2 ... vs. the XeTeX situation

XeTeX was primarily written to extend ε-TeX with full Unicode support, as well as loading system fonts. Its development started from ε-TeX, rather than from pdfTeX, which had added various new primitives on top of ε-TeX. Many of pdfTeX's additions to ε-TeX have to do with directly producing PDF output (ε-TeX supports only DVI), but others are entirely independent of that.

Over the years, some of these other "utilities" have been added to XeTeX (for example `\pdfstrcmp`, which in XeTeX is just `\strcmp`). However, several have not made it, but *have* been added to pTeX and upTeX. That has meant that XeTeX has been "a bit behind" in feature terms: some things simply can't be done without primitive support.

3 A development opportunity arises

Recently, a Travis-CI testing environment has been created for TeX Live (see `github.com/TeX-Live/texlive-source`), meaning that it's now *easy* to try adding new material to the `WEB` sources of pdfTeX, XeTeX, etc. As part of more general work on primitives, it made sense to bring XeTeX "back in line" with (u)pTeX. That's important for expl3, as the LaTeX team have been using almost all of the primitives that were "missing" in XeTeX, as well wanting to bring `\expanded` into the mainstream.

4 Providing `\expanded`

For some time, the LaTeX team have been thinking about asking for `\expanded` to be made more widely available. Unlike the `\romannumeral` "trick", `\expanded` does not require any hard work to get "past" any output, so it is very useful for creating macros that work like functions. It's also fast and clear in intention.

The code itself was easy enough to move around: a bit of copy-pasting! As well as merging into the stable branch of pdfTeX, I worked out how to add `\expanded` to XeTeX and the Japanese TeX engines pTeX and upTeX. So soon we'll all be able to do

```
\def\a{\b}\def\b{c}
\message{Hello \a\space #}
\detokenize\expandafter
  {\expanded{Hello \a\space #}}
\bye
```

(Try the example in LuaTeX if you don't have the burning edge pdfTeX binaries.)

5 New primitives in XeTeX

So, besides `\expanded`, what has been added? The new additions are all named without the `pdf` prefix that pdfTeX includes, as they have nothing to do with PDFs (and XeTeX is not pdfTeX):

```
\creationdate   \elapsedtime   \filedump
\filemoddate    \filesize      \resettimer
\normaldeviate  \uniformdeviate \randomseed
```

These enable things like random numbers in the LaTeX3 FPU, measuring code performance, and checking the details of files: all stuff that is in expl3 will now work with XeTeX.

I should add that although I did the grind of working out how to integrate the pdfTeX code into XeTeX, Akira Kakuto sorted out the areas that needed knowledge of C, in particular where XeTeX's Unicode internals don't match up with pdfTeX's 8-bit ones.

6 Adjusting `\Ucharcat`

I made one other minor adjustment to XeTeX: altering how `\Ucharcat` works so it can create category code 13 ("active") tokens. That probably won't show up for users; however, it helps the team extend some low-level expl3 code. It should just mean one fewer XeTeX restriction.

7 Getting the code

TeX Live gets binary updates only once per year, so users there will need to wait for the 2019 release. On the other hand, MiKTeX already has the new features, so if you are on Windows it's pretty trivial to try. If you use TeX Live and want to test this out, you can update your binaries in-place, for example from W32TeX (`w32tex.org`): if you understand what that means, you probably know how to do it!

◇ Joseph Wright
Northampton, United Kingdom
joseph dot wright (at)
morningstar2.co.uk