What's new in the XeTeX world

Jonathan Kew

jonathan_kew@sil.org

SIL International

May 2, 2008

Looking back over the year

- TeX Live 2007: XeTeX 0.996
- some distros used "0.996-patch1" with several bug-fixes
- code development continued (gradually) through the year
  - updated OpenType and Unicode support libraries
  - OpenType math support
  - added Graphite font support
  - synthetic font styles
  - several new primitives (inter-char tokens, count PDF pages)
  - output driver bug-fixes
  - some distros shipped development snapshots with an "0.997" version number (e.g., MikTeX 2.7), though this was not officially released
  - several macro packages now depend on post-0.996 features
- recently tagged code for version 0.998, which will be the TeX Live 2008 version

OpenType layout support

- ICU library updated from release 3.4 to 3.8.1
- some post-3.8.1 patches also adopted from ICU's repository
- support for additional scripts (e.g., Sinhala)
- character repertoire updated to Unicode 5
- XeTeX extensions to ICU layout engine:
  - user specification of optional features
  - passing parameters to features such as `salt`, `ssty`
  - additional script support (Syriac, Thaana, basic Mongolian)

Note for packagers/distributors: because of the ICU layout API extensions used, XeTeX cannot be built using the platform's standard ICU libraries; it must be linked with its own private build (as found in the source tree).

Graphite font support

First demonstrated at EuroBachoTEX 2007, now integrated into the TEX Live repository and will be available in the TL2008 release.

Graphite[1] is an SIL project to provide rendering capabilities for complex non-Roman writing systems, including scripts that are not yet standardized in Unicode, or require behaviors that are not supported in the OpenType model.

Not required for most major languages or scripts, but for the minority communities who need it, this may be the only available technology (besides pencil and paper) that handles their writing system correctly. Graphite implementations currently in use with XƎTEX include minority-language extensions to Burmese, and the N'Ko script from West Africa.

[1] http://scripts.sil.org/RenderingGraphite

Synthetic font styles

When using OpenType or TrueType fonts directly, without `.tfm` files, `.map` files, etc., XeTeX originally supported only styles for which actual physical fonts are available.

There is now also support for creating "synthetic" styles when necessary. These are specified as part of the font name, like optional OpenType features:

- slant=*n* — *slanted* and reverse slanted
- extend=*n* — extended and compressed
- embolden=*n* — **emboldened**
- letterspace=*n* — letterspaced

CJK support packages

The “inter-character tokens” feature (demonstrated last year at EBT2007) has allowed CJK users to create macro packages that handle many of the refinements needed for proper typesetting of these languages:

- `jspacing.sty` – see archives of XeTeX mailing list
- `zhspacing.sty` – see Google Code
- `xeCJK` – see `ctex.org` (site in Chinese)

The `polyglossia` package

Package created by François Charette, following discussions on the XeTeX mailing list, to provide a modern, Unicode-friendly alternative to Babel.

- selection of hyphenation patterns for the current language
- setting language-dependent strings
- modules for alphanumerical number notations and for calendrical computations
- language-specific typographic conventions such as French punctuation spacing, using the inter-character token insertion feature
- directionality for right-to-left languages
- interaction with script and language features of fonts (via `fontspec`)

Unicode normalization

One of the complications of Unicode is that there are often multiple ways to encode the same text, particularly in the area of accented letters. Thus, 'ä' could be encoded in two ways:

> *U+00E4 LATIN SMALL LETTER A WITH DIAERESIS*
> *U+0061 LATIN SMALL LETTER A, U+0308 COMBINING DIAERESIS*

Unicode defines these to be *canonical equivalents*, which are interchangeable and should appear and function identically.

In an ideal world, all fonts would support both the precomposed characters and the combining diacritics. However, this is often not the case; many fonts only support the precomposed characters, and will give inferior results or even their `.notdef` glyph if presented with the combining diacritics.

Unicode normalization

Canonical equivalence also raises questions for XeTeX itself: Is `\mäcro` written with the precomposed *a-diaeresis* the same control sequence as `\mäcro` written with the character sequence *a, combining diaeresis*?

To help address these issues, XeTeX 0.998 supports a new parameter `\XeTeXinputnormalization`. If this is set to 1, all input text will be *normalized* according to the Unicode NFC definition (roughly speaking, using precomposed characters wherever possible) as it is read from the file. If it is set to 2, all input text will be normalized as NFD (using sequences with combining marks rather than precomposed accented letters).

In most cases, normalizing input text to NFC will give the best compatibility with a wide range of fonts, so I recommend this for normal use (and will probably make it the default).

SyncTEX support

SyncTEX is a new extension for the various TEX-based engines, written by Jérôme Laurens with input from other developers. Like the older `-source-specials` feature available in some TEX implementations and previewers, or the `pdfsync` macro package, it provides support for navigating between the source text and the typeset document.

The basic mechanism is that the various nodes within TEX's data structures are enlarged and each is "stamped" with a reference to the source file and line that created it. This data, along with the node's final position on the page, is written to the `.synctex` file and provides the information needed for previewers or other tools to provide synchronization.

XeTeX and other engines

The two key features XeTeX offers are (a) native support for Unicode, including complex non-Latin scripts, and (b) easy use of modern font formats (TrueType and OpenType).

Earlier, Unicode support was offered by Omega (and then Aleph); more recently, this has been incorporated into LuaTeX, which also has support for direct use of OpenType fonts. Nevertheless, there are major differences in the approach taken by the different projects. A few thoughts:

XeTeX values

- ease of setup and use
- use of available libraries wherever feasible
- do "the right thing" automatically

LuaTeX (and predecessors)

- ultimate flexibility
- control every aspect of the implementation
- provide authors or macro writers with low-level tools