Another incarnation of Lucida: Lucida + Lucida Math OpenType

Mojca Miklavec, Sežana, Slovenia
Ulrik Vieth, Stuttgart, Germany

EuroBachoTeX 2011
Brodnica, Poland, 2011
Overview of this talk

- Historical perspective (Choices of available fonts)
- History and Background of Lucida fonts
- Scope of the Lucida TUG distribution
- Problems of the Lucida TUG distribution
- Goals for the Lucida OpenType project
- Project Organization and Team
- Details of font development
- Scope of the Lucida OpenType distribution
- Project Timeline and Status
Historical Perspective

• \TeX{} has been around for 30+ years
• \TeX{} has gone through 3 eras of font technology
  • MetaFont fonts (1980s–1990s)
  • PostScript fonts (1990s–2000s)
  • OpenType fonts (2000s–current)
• Choices of text fonts have grown very large
  • any Type 1 text fonts can be set up for use with pdf\TeX{}
  • any OpenType text fonts can be used with (Lua|Xe)\TeX{}
• Choices of math fonts have always been limited
  • few Type 1 math fonts exist (and require special setup)
  • few OpenType math fonts exist and are ready to use
• Choices of complete families are also limited
  • few complete families exist (Serif, Sans, Mono, Math)
Choices of available math fonts (I)

- MetaFont era (1980s–1990s)
  - CM, AMS Symbols, AMS Euler (original MF designs)
  - Concrete Math, CM Bright (late additions, 1995–2000)
- PostScript era (1990s–2000s)
  - CM, AMS Symbols, AMS Euler (converted from MF)
  - Times + MathTime (sold commercially, early 1990s)
  - Lucida Bright + Math (sold commercially, early 1990s)
  - TM-Math, HV-Math, etc (sold commercially, early 2000s)
  - TX/PX Fonts, MathPazo (late additions, 2000–2005)
  - Fourier, MathDesign (late additions, 2005–2010)

- Why does Lucida matter?
  - Lucida is one of the few available math fonts
  - Lucida is one of the few complete font families
Choices of available math fonts (II)

- OpenType era (2000s–current)
  - Cambria Math (original OT math reference font)
  - Asana Math (converted from PS, derived from PX)
  - Times + XITS Math (derived from STIX, 2000–2010)
  - Neo Euler (under development, incomplete)
  - Lucida Bright + Math (under development)
  - Latin Modern + LM Math (under development)
  - \TeX Gyre + TG Math (under development)

- Why does Lucida matter?
  - Choices of OpenType math fonts now in 2010 are as limited as PS math fonts in early 1990s
  - Choices of OpenType math fonts are needed to attract users to new Unicode \TeX engines
History of Lucida fonts (I)

- Developed by Bigelow & Holmes
  - Chuck Bigelow: professor at Stanford, Rochester RIT
  - Kris Holmes: type designer of 100+ typefaces
- Design goals
  - usable at low-resolutions (high x-height, high readability)
  - complete extended font family (Serif, Sans, Mono, etc)
  - suitable for math (Greek, symbols, arrows, dingbats, etc)
- Development history
  - originally designed in mid/late 1980s
  - revised and adjusted for \TeX in early 1990s
- Available Distributions
  - original version sold via various font companies
  - revised version sold via Y\&Y Inc. (early 1990s-2000s)
  - current versions sold via TUG and PCTeX Inc.
History of Lucida fonts (II)

- additional Lucida Unicode distributions exists
  - LucidaConsole, Lucida Sans Unicode (MS Windows)
  - Lucida Grande (Apple Mac OS X)
  - Lucida Bright/Sans/Typewriter (Sun Java JDK)

- Limitations of TUG / Y&Y distribution (Type 1)
  - full set of families (including math and extras)
  - only available in Type 1 format (8-bit font sets)
  - difficult to set up and use (font encoding mess)
  - limited character set / symbol set

- Limitations of Lucida Unicode distributions (TTF)
  - extended character set (Unicode BMP)
  - limited set of families (no math, no extras)
Scope of the Lucida TUG distribution

• Members of Lucida font families (Type 1)
  • Lucida Bright (serif)
  • Lucida New Math
  • Lucida Sans Typewriter
  • Lucida Sans
  • Lucida Typewriter (serif)
  • Lucida Fax, Lucida Casual
  • Lucida Blackletter, Calligraphic, Handwriting

• bundled in two packages
  • basic set + expert set (Y&Y)
  • basic set + complete set (TUG)

• Coverage of font sets
  • text: one 8-bit set each, either TeXnANSI (LY1) or similar
  • math: several 8-bit sets, about same as CM + AMS + extras
Existing Fonts: TUG’s Type1 (I)

- Lucida Bright
  - LucidaBright + SMALLCAPS
  - LucidaBright-Oblique
  - LucidaBright-Italic
  - LucidaBright-Demi + SMALLCAPS
  - LucidaBright-DemiItalic
- Lucida Sans Typewriter
  - LucidaSans-Typewriter
  - LucidaSans-TypewriterOblique
  - LucidaSans-TypewriterBold
  - LucidaSans-TypewriterBoldOblique
- Lucida New Math
  - ... many fonts ...
Existing Fonts: TUG’s Type1 (II)

- Lucida Sans
  - LucidaSans
  - LucidaSans-Italic
  - LucidaSans-Demi
  - LucidaSans-DemiItalic
  - LucidaSans-Bold
  - LucidaSans-BoldItalic
- Lucida Typewriter
  - LucidaTypewriter
  - LucidaTypewriterOblique
  - LucidaTypewriterBold
  - LucidaTypewriterBoldOblique
Existing Fonts: TUG’s Type1 (III)

- Lucida Fax
  - LucidaFax
  - *LucidaFax-Italic*
  - LucidaFax-Demi
  - *LucidaFax-DemiItalic*
- Lucida Casual
  - LucidaCasual
  - *LucidaCasual-Italic*
- Lucida Blackletter
  - LucidaBlackletter
- Lucida Calligraphy
  - *LucidaCalligraphy-Italic*
- Lucida Handwriting
  - *LucidaHandwriting-Italic*
Problems of the Lucida TUG distribution

(a) old: virtual (faked) glyph:
   - wrong vertical position
   - centered horizontally – also wrong

(b) new: properly positioned accent

(c) old & new: existing glyph
Problems of the Lucida TUG distribution

- Problems / Limitations of 8-bit text fonts
  - limited or incomplete language support in base fonts
  - no direct support for T1 (Cork) encoding by Y&Y
  - support for T1 (Cork) only via virtual fonts (fontinst)
  - some glyphs faked, some glyphs missing or broken
  - no support for other encodings at all (CS, QX, L7X, etc)
Goals for the Lucida OpenType project

• So one day the idea for an new project was born:
  • *Let’s create a new version of Lucida OpenType!*

• Development goals
  • Repackage Type 1 fonts in OpenType fonts
  • Extend the coverage of the OpenType fonts
  • Provide good Unicode language support (Latin)
    (Design additional accented glyphs as needed)
  • Provide good Unicode math support
    (Design additional math symbols/alphabets)
  • Use latest OpenType font technology
    (Create a full-featured OpenType math font)
Project Organization and Team

- Project Organization
  - Project set up by TUG in cooperation with B&H
  - B&H provided the designs of additional glyphs
  - Khaled Hosny did the development (supported by TUG)
  - Team members did the testing (and will get free fonts)
  - Fonts will eventually be sold via TUG and B&H

- Team Members
  - Karl Berry (TUG): coordination, legal stuff
  - Chuck Bigelow, Kris Holmes (B&H): glyph design
  - Khaled Hosny: font technology, assembly, metrics
  - Mojca Miklavec: testing of text fonts / languages
  - Ulrik Vieth: testing of math fonts / symbols
  - Hans Hagen, Taco Hoekwater: technical advisory
Positioning of combining marks (I)
Positioning of combining marks (II)

StartChar: uni030C
Encoding: 780 780 354
Width: 0
Flags: W
AnchorPoint: "MarkAbove" -301 626 mark 0
SplineSet
-355 626 m 1
-463 771 l 1
-421 771 l 1
-301 675 l 1
-180 771 l 1
-138 771 l 1
-247 626 l 1
-355 626 l 1
EndSplineSet
EndChar
Positioning of combining marks (III)

StartChar: ccaron
Encoding: 269 269 374
Width: 509
VWidth: 0
AnchorPoint: "MarkBelow" 303 -96 basechar 0
Refer: 354 780 N 1 0 0 1 604 0 2
Refer: 85 99 N 1 0 0 1 0 0 2
EndChar
Combining accents (I)

- Work properly when using positioning marks:
  - **mark**: MarkToBase font feature
  - **mkmk**: MarkToMark

- **Examples**:
  - c + combining caron (U+030C)
    - properly: \v{c}\v{c} \quad LM-like: \v{c}\v{c} \quad T\!E\!X: \v{c}\v{c}
  - U + combining inverted breve below (U+032F)
    - properly: \u{u}\u{u} \quad LM-like: \u{u}\u{u}
Combining accents (II)

- Font features: **mark** (MarkToBase) & **mkmk** (MarkToMark)
- Ideally requires placing top and bottom position marks to all letters: time consuming
- Very tight time constraints:
  - finished only in Lucida Bright Regular
  - incomplete in other fonts (font features are disabled)
  - high priority to be completed
- It would be nice if also **OpenType Latin Modern** followed the standard.
Scope of Lucida OpenType distribution

- Members of Lucida font families (OTF)
  - Lucida Bright (serif)
  - Lucida Math
  - Lucida Sans Typewriter
- Coverage of text fonts
  - Latin: most European languages covered
  - Greek: not in scope, only as needed for math
  - Cyrillic, Arabic, Hebrew, Thai: not in scope
- Coverage of math alphabets
  - most math alphabets covered (few exceptions)
  - most math symbols covered
Scope of Lucida OpenType distribution

• Some numbers:
  • other Lucida Unicode fonts:
    • Lucida Console: 667 total glyphs, 244 lowercase
    • Lucida Sans Unicode: 1779 total glyphs, 465 lowercase
    • Lucida Grande: 2826 total glyphs, 1072 lowercase
  • old Lucida Type1 fonts:
    • Lucida Bright: 252 glyphs (including ff-ligs)
    • Lucida Sans TT: 249 glyphs
  • new Lucida OTF fonts:
    • Lucida Bright (rm): 956 total glyphs, 476 lowercase (incl. sc)
    • Lucida Bright (bf): 528 total glyphs, 230 lowercase (incl. sc)
    • Lucida Bright (it): 396 total glyphs, 138 lowercase
    • Lucida Bright (bi): 396 total glyphs, 138 lowercase
    • Lucida Sans TT: 359 total glyphs, 120 lowercase
New OTF fonts in the TUG distribution

- Lucida Bright
  - LucidaBright
  - *LucidaBright-Italic*
  - LucidaBright-Demi
  - *LucidaBright-DemiItalic*

- Lucida Sans Typewriter
  - LucidaSans-Typewriter
  - *LucidaSans-TypewriterOblique*
  - LucidaSans-TypewriterBold
  - *LucidaSans-TypewriterBoldOblique*

- Lucida Math
  - ... only one font ...
Alphabets in the math fonts

- Math alphabets

<table>
<thead>
<tr>
<th>Alphabets</th>
<th>ABCXYZ abcxyz</th>
<th>ΑΒΓΞΨΩ αβγξψω</th>
<th>0123</th>
</tr>
</thead>
<tbody>
<tr>
<td>\mathup</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbfup</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbfit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathsfup</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathsfit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbfsfup</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbfsfit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathscr</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbfsccr</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathfrak</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbfrak</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>\mathbbb</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- \mathup
- \mathit
- \mathbfup
- \mathbfit
- \mathsfup
- \mathsfit
- \mathbfsfup
- \mathbfsfit
- \mathscr
- \mathbfsccr
- \mathfrak
- \mathbfrak
- \mathbbb
Samples of math fonts

• Math sample

**Theorem 1 (Residue Theorem).** Let \( f \) be analytic in the region \( G \) except for the isolated singularities \( a_1, a_2, \ldots, a_m \). If \( \gamma \) is a closed rectifiable curve in \( G \) which does not pass through any of the points \( a_k \) and if \( \gamma \approx 0 \) in \( G \) then

\[
\frac{1}{2\pi i} \int_\gamma f = \sum_{k=1}^m n(\gamma; a_k) \text{Res}(f; a_k).
\]

**Theorem 2 (Maximum Modulus).** Let \( G \) be a bounded open set in \( \mathbb{C} \) and suppose that \( f \) is a continuous function on \( G^− \) which is analytic in \( G \). Then

\[
\max\{|f(z)| : z \in G^−\} = \max\{|f(z)| : z \in \partial G\}.
\]
Project Timeline and Status

• Project Timeline
  • Project idea in September 2010 (ConTeXt meeting)
  • Project set up by TUG with B&H in November 2010
  • Glyph designs provided by B&H in December 2010
  • Font development and testing since January 2011

• Project Status:
  • few missing letters in text fonts to be designed
  • few missing symbols in math fonts to be designed
  • few missing math alphabets (e.g. bold Script, Fraktur)
  • more testing needed to ensure quality / compatibility
  • Font development and testing will continue
  • Khaled will have to leave the project very soon
  • B&H have no time now to draw remaining glyphs now
  • remaining work may be delayed, no release date yet