OpenType Math Illuminated

Dr. Ulrik Vieth

Stuttgart, Germany

BachoTEX 2009
Developments in text typesetting

- Major trends in publishing
  - support for Unicode character sets
  - support for OpenType font technology
- Major developments in the \TeX{} community
  - new \TeX{} engines: \texttt{Xe\TeX}, \texttt{Lua\TeX}
  - new \TeX{} fonts: Latin Modern, \TeX{} Gyre
- Outside developments
  - OpenType supported by operating systems or libraries
  - OpenType supported by typesetting software
  - OpenType supported by commercial font suppliers
  - OpenType as a replacement for TrueType and Type 1
Developments in math typesetting

- **Unicode math**
  - encoding for math symbols and alphabets
  - developed by working group (input from STIX, AMS)
  - standard since 2001 (UTR#25 for Unicode 3.2)

- **OpenType math**
  - extension of OpenType font format
  - developed by Microsoft as a vendor-controlled format
  - officially *experimental*, but already *de facto* standard
  - first implemented in MS Office 2007
  - supported by reference fonts: Cambria Math
  - supported by font editors and tools: FontForge
  - supported by new \TeX\ engines: X\LaTeX, Lua\TeX
Overview of OpenType math

• OpenType font format
  • extensible table structure (as in TrueType)
  • different flavors of font outlines (TrueType vs. CFF)
  • some tables required, e.g. glyph metrics, outlines
  • some tables optional, e.g. advanced typographic features
  • additions for OpenType math: new optional MATH table

• OpenType MATH table
  • global font parameters (similar to fontdimens of Appendix G)
  • variants and constructions (similar to charlists and extensibles)
  • additions to glyph metrics (similar to overloaded TFM fields)
Interactive Demo (I)

- Interactive Demo
  - open Cambria Math in FontForge
  - inspect parameters of MATH table
Font parameters

- Font parameters in TeX math fonts
  - approx. 20 parameters explicit in font metrics
  - many parameters implicit in typesetting algorithms
  - some parameters hidden in macro definitions
- Font parameters in OpenType math fonts
  - approx. 60 parameters explicit in MATH table
  - most TeX parameters have clear correspondence
  - some TeX parameters have no correspondence
  - some extensions / generalizations of TeX concepts
  - some cleanup of overloaded font data structures
Big Operators

- Spacing of limits on big operators
  - 5 parameters in \TeX\ fontdimens
  - 4 parameters in OT MATH table
  - clear correspondence for $\xi_9$ to $\xi_{12}$
  - no correspondence for $\xi_{13}$
  - outside clearance assumed zero

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>$\xi$</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpperLimitBaselineRiseMin</td>
<td>$\xi_{11}$</td>
</tr>
<tr>
<td>UpperLimitGapMin</td>
<td>$\xi_9$</td>
</tr>
<tr>
<td>LowerLimitGapMin</td>
<td>$\xi_{10}$</td>
</tr>
<tr>
<td>LowerLimitBaselineDropMin</td>
<td>$\xi_{12}$</td>
</tr>
</tbody>
</table>
Stretch Stacks

- Spacing of stretch stacks
  - generalization of stacked elements
  - e.g. labels above/below arrows
  - e.g. over/underbraces on formulas
  - correspondence at macro level in \( \text{T}_{\text{EX}} \)
  - spacing similar to big operators

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>StretchStackTopShiftUp</td>
<td>( \xi_{11} )</td>
</tr>
<tr>
<td>StretchStackGapAboveMin</td>
<td>( \xi_{9} )</td>
</tr>
<tr>
<td>StretchStackGapBelowMin</td>
<td>( \xi_{10} )</td>
</tr>
<tr>
<td>StretchStackBottomShiftDown</td>
<td>( \xi_{12} )</td>
</tr>
</tbody>
</table>
Over- and Underlines

• Spacing of over- and underlines
  • 6 built-in rules in \texttt{TEX} algorithms
  • 6 parameters in OT MATH table
  • implicit rules made explicit in OT
  • greater flexibility of font designer

• Parameter mapping

\begin{tabular}{ll}
  OverbarExtraAscender & (= \xi_8) \\
  OverbarRuleThickness & (= \xi_8) \\
  OverbarVerticalGap & (= 3 \xi_8) \\
  UnderbarVerticalGap & (= 3 \xi_8) \\
  UnderbarRuleThickness & (= \xi_8) \\
  UnderbarExtraDescender & (= \xi_8) \\
\end{tabular}
Fractions and Stacks (I)

- Spacing of regular fractions
  - 4 parameters in \TeX\ fontdimens
  - 5 built-in rules in \TeX\ algorithms
  - 9 parameters in OT MATH table

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>(\sigma_8) styles (D, D')</th>
<th>(\sigma_9) other styles</th>
<th>(\sigma_{11}) styles (D, D')</th>
<th>(\sigma_{12}) other styles</th>
</tr>
</thead>
<tbody>
<tr>
<td>FractionNumeratorDisplayStyleShiftUp</td>
<td>(\sigma_8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionNumeratorShiftUp</td>
<td>(\sigma_9)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionNumeratorDisplayStyleGapMin</td>
<td>(= 3 \xi_8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionNumeratorGapMin</td>
<td>(= \xi_8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionRuleThickness</td>
<td>(= \xi_8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionDenominatorDisplayStyleGapMin</td>
<td>(= 3 \xi_8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionDenominatorGapMin</td>
<td>(= \xi_8)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionDenominatorDisplayStyleShiftDown</td>
<td>(\sigma_{11})</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FractionDenominatorShiftDown</td>
<td>(\sigma_{12})</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Fractions and Stacks (II)

- Spacing of generalized fractions (stacks)
  - 4 parameters in \( \TeX \) fontdimens (overlap between fractions and stacks)
  - 2 built-in rules in \( \TeX \) algorithms
  - 6 parameters in OT MATH table (no overlap between fractions and stacks)

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>( \sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>StackTopDisplayStyleShiftUp</td>
<td>( \sigma_8 )</td>
</tr>
<tr>
<td>StackTopShiftUp</td>
<td>( \sigma_{10} )</td>
</tr>
<tr>
<td>StackDisplayStyleGapMin</td>
<td>( = 7 \xi_8 )</td>
</tr>
<tr>
<td>StackGapMin</td>
<td>( = 3 \xi_8 )</td>
</tr>
<tr>
<td>StackBottomDisplayStyleShiftDown</td>
<td>( \sigma_{11} )</td>
</tr>
<tr>
<td>StackBottomShiftDown</td>
<td>( \sigma_{12} )</td>
</tr>
</tbody>
</table>
Superscripts and Subscripts (I)

- Spacing of superscripts and subscripts
  - 7 parameters in \( \text{T}_{\text{E}}\text{X} \) fontdimens
  - 5 parameters in OT MATH table
  - no distinction between \( \sigma_{13} \) and \( \sigma_{14} \) (superscripts in display or text style)
  - no distinction between \( \sigma_{16} \) and \( \sigma_{17} \) (subscripts with or w/o superscripts)

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>( \sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuperscriptShiftUp</td>
<td>( \sigma_{13}, \sigma_{14} )</td>
</tr>
<tr>
<td>SuperscriptShiftUpCrammed</td>
<td>( \sigma_{15} )</td>
</tr>
<tr>
<td>SubscriptShiftDown</td>
<td>( \sigma_{16}, \sigma_{17} )</td>
</tr>
<tr>
<td>SuperscriptBaselineDropMax</td>
<td>( \sigma_{18} )</td>
</tr>
<tr>
<td>SubscriptBaselineDropMin</td>
<td>( \sigma_{19} )</td>
</tr>
</tbody>
</table>
Superscripts and Subscripts (II)

- Spacing when resolving collisions between superscripts and subscripts
  - 4 built-in rules in TeX algorithms
  - 4 parameters in OT MATH table
  - implicit rules made explicit in OT

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuperscriptBottomMin</td>
<td>( \frac{1}{4} \sigma_5 )</td>
</tr>
<tr>
<td>SubscriptTopMax</td>
<td>( \frac{4}{5} \sigma_5 )</td>
</tr>
<tr>
<td>SubSuperscriptGapMin</td>
<td>( 4 \xi_8 )</td>
</tr>
<tr>
<td>SuperscriptBottomMaxWithSubscript</td>
<td>( \frac{4}{5} \sigma_5 )</td>
</tr>
</tbody>
</table>
Radicals (I)

- Spacing of radicals (square roots)
  - 4 built-in rules in \TeX{} algorithms
  - 4 parameters in OT MATH table
  - unusual metrics in \TeX{}: $h\sqrt{} = \xi_8$
  - no need for unusual metrics in OT

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>RadicalExtraAscender</td>
<td>$= \xi_8$</td>
</tr>
<tr>
<td>RadicalRuleThickness</td>
<td>$= h\sqrt{} = \xi_8$</td>
</tr>
<tr>
<td>RadicalDisplayStyleVerticalGap</td>
<td>$= \xi_8 + \frac{1}{4}\sigma_5$</td>
</tr>
<tr>
<td>RadicalVerticalGap</td>
<td>$= \xi_8 + \frac{1}{4}\xi_8$</td>
</tr>
</tbody>
</table>
Radicals (II)

- Spacing of radicals ($n$-th roots)
  - 3 parameters hidden in \TeX{} macros
  - 3 parameters in OT MATH table
  - implicit rules made explicit in OT
  - replacement of macros by primitives

- Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>RadicalKernBeforeDegree</td>
<td>e.g. 5/18 em</td>
</tr>
<tr>
<td>RadicalKernAfterDegree</td>
<td>e.g. 10/18 em</td>
</tr>
<tr>
<td>RadicalDegreeBottomRaisePercent</td>
<td>e.g. 60%</td>
</tr>
</tbody>
</table>
General parameters

• Mixed bag of parameters
  • some related to font sizes of script fonts
  • some related to size of delimited fractions
  • some related to placement of math accents

• Parameter mapping

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScriptPercentScaleDown</td>
<td>e.g. 70–80 %</td>
</tr>
<tr>
<td>ScriptScriptPercentScaleDown</td>
<td>e.g. 50–60 %</td>
</tr>
<tr>
<td>DisplayOperatorMinHeight</td>
<td>?? (e.g. 12–15 pt)</td>
</tr>
<tr>
<td>DelimitedSubFormulaMinHeight</td>
<td>σ₂₀ (e.g. 20–24 pt)</td>
</tr>
<tr>
<td>AxisHeight</td>
<td>σ₂₂ (axis height)</td>
</tr>
<tr>
<td>AccentBaseHeight</td>
<td>σ₅ (x-height)</td>
</tr>
<tr>
<td>FlattenedAccentBaseHeight</td>
<td>?? (capital height)</td>
</tr>
</tbody>
</table>
Font sizes of script fonts (I)

- In \TeX\ math fonts
  - math families always loaded at 3 font sizes
  - font sizes of script fonts defined outside the font
  - font sizes defined in macro packages or format files

- Example (Computer Modern, using optical design sizes)
  \begin{verbatim}
  \newfam\symbols
  \textfont\symbols=cmsy10 \% at 10.0 pt
  \scriptfont\symbols=cmsy7 \% at 7.0 pt
  \scriptscriptfont\symbols=cmsy5 \% at 5.0 pt
  \end{verbatim}

- Example (Math Times, using scaled-down sizes)
  \begin{verbatim}
  \newfam\symbols
  \textfont\symbols=mtsy10 \% at 10.0 pt
  \scriptfont\symbols=mtsy10 scaled 760 \% at 7.6 pt
  \scriptscriptfont\symbols=mtsy10 scaled 600 \% at 6.0 pt
  \end{verbatim}
Font sizes of script fonts (II)

- **In OpenType math fonts**
  - font sizes of script fonts specified inside the font
  - optical variants for script sizes packaged in base font
  - optical variants activated by OpenType feature tags
  - potentially only a single OpenType math font needed
  - math families can still be loaded at 3 font sizes

- **Example (using scaling factors and features)**

  \begin{verbatim}
  \newfam\symbols
  \textfont\symbols=\textfont\textfont=CambriaMath \% at 10.0 pt
  \scriptfont\symbols=\textfont\scriptfont=CambriaMath:+ssty0
      \textfont\textfont=scaled \OTvalue{ScriptPercentScaleDown}
  \scriptscriptfont\symbols=\textfont\scriptscriptfont=CambriaMath:+ssty1
      \textfont\textfont=scaled \OTvalue{ScriptScriptPercentScaleDown}
  \end{verbatim}
Delimited Fractions

- **What’s a delimited fraction?**
  - $$\left( {n \atop k} \right)$$ (regular fraction)
  - $$\{n \atopwithdelims() k\}$$ (delimited fraction)

- **What’s the difference?**
  - size depends on delimiterfactor, delimitershortfall
    ⇒ 18 pt or 24 pt delimiters (depending on contents)
  - size depends on fontdimens $\sigma_{20}$ (display), $\sigma_{21}$ (text)
    ⇒ always 24 pt delimiters (regardless of contents)

- **What’s the problem?**
  - only one OpenType parameter DelimitedSubFormulaMinHeight
  - no suitable correspondence for \TeX parameters $\sigma_{20}$, $\sigma_{21}$
  - no suitable implementation for \atopwithdelims
Variants and Constructions

• In \TeX\ math fonts:
  • charlists and extensibles only used in specific contexts
  • big operators: 2 vertical sizes (text style vs. display style)
  • big delimiters: $n$ vertical sizes + extensible version
  • wide accents: $n$ horizontal sizes, but no extensible version

• In OpenType math fonts:
  • generalization of variants and constructions
  • big operators: can have more than 2 vertical sizes
  • big operators: can even have extensible version
  • wide accents: can also have extensible version
  • wide accents: can be applied to overbrace/underbrace
  • long arrows: can be represented by horizontal constructions
Big Operators

- In $\TeX$ math fonts:
  - only 2 sizes of operators (text style vs. display style)
  - no support for additional sizes or extensible versions
- In OpenType math fonts:
  - possible to have additional sizes of operators
  - OpenType parameter \texttt{DisplayOperatorMinHeight} needed to determine which size to use in display style
  - possible to have extensible versions of operators (depends on glyph shape, e.g. straight integral)
  - semantics may be difficult to implement in $\TeX$ (need context to determine size of operators)
  - semantics may be easier to implement in MathML
Big Delimiters

- In \TeX\ math fonts:
  - usually 4 sizes of delimiters + extensible version
  - usual progression of sizes: 12 pt, 18 pt, 24 pt, 30 pt
  - macros to select specific sizes: big, Big, bigg, Bigg
  - no requirement to have 4 sizes, just a convention

- In OpenType math fonts:
  - possible to have additional or intermediate sizes, e.g. 4 of the usual sizes + 3 intermediate sizes
  - no limitations such as 16 TFM heights/depths
  - only base size of delimiters encoded in Unicode slots
  - additional sizes encoded in private-use area using internal glyph names symbol.vsize<n> or symbolbig<n>
Wide Accents

- In TeX math fonts:
  - only limited range of wide accents provided in fonts
  - no support for extensible versions of math accents
  - macro constructions used as a workaround (leaders)

- In OpenType math fonts:
  - possible to have extensible versions of math accents
  - possible to rewrite/simplify macro constructions
  - possible to redefine overbrace/underbrace as math accents (may require different semantics for labels on braces)
  - only base size of math accents encoded in Unicode slots
  - additional sizes encoded in private-use area using internal glyph names `symbol.hsize<n>` or `symbolwide<n>`
Interactive Demo (II)

- Interactive Demo
  - open Cambria Math in FontForge
  - inspect variants and constructions
Font parameters

- **OpenType MATH** extends many **TEX** concepts
  - many built-in rules replaced by explicit parameters
  - some overlap in multi-purpose parameters avoided
  - some macro parameters integrated (e.g. degree of radicals)
  - some extensions of concepts integrated (e.g. stretch stacks)

- **OpenType MATH** falls short on a few **TEX** concepts
  - outside clearance on big operators ($\xi_{13} = 0$)
  - superscripts in display or text style ($\sigma_{13} \neq \sigma_{14}$)
  - subscripts with or w/o superscripts ($\sigma_{16} \neq \sigma_{17}$)
  - nothing suitable for delimited fractions ($\sigma_{20}, \sigma_{21}$)

- **OpenType MATH** cannot reproduce 100% of **TEX** behavior
- **TEX** engines can add the missing bits, if really needed
Variants and Constructions

- **OpenType MATH** extends many **\TeX** concepts
  - generalization of applicable context
  - additional sizes + extensible versions of big operators
  - additional sizes + extensible versions of wide accents
  - horizontal constructions not limited to math accents, also applicable for long arrows or over/under delimiters

- **\TeX** engines may need to implement new semantics
  - big operators may need context to determine size
  - new primitives needed for over/under delimiters
  - new primitives needed for labels on long arrows
  - macros can be rewritten/simplified using new primitives