Authoring Mathematical Knowledge

Bruce R. Miller
NIST

Digital Library of Mathematical Functions

2 Issues

- Extracting mathematical content
  (formula level)
  Challenging enough. . .
- Determining it's role & interrelations.
  (document level)
  Useful for validation.
  Hope to learn more here.

Authoring for MKM: Ideal?

- Integrate document creation with derivation/proof.
- Valid, machine readable

But, for DLMF

- Are we there yet?
- Dozens of authors using compatible tools?
- What they *know* vs. willing to *prove*?

Digital Library of Mathematical Functions

Authoring for MKM: LaTeX?

- A good choice. . .
  - Author familiarity, convenience (for some)
  - Logical document structure (sort of).
  - Expressive for mathematics
  - Beautiful typography!
- and a bad one.
  - Needs more structure
  - Quirky computational model
  - Ambiguous math markup

Digital Library of Mathematical Functions

LaTeXML Goals

- LaTeX ⇒ XML Transformer
  - General purpose
  - LaTeX-like DTD (or other?)
  - Math to MathML, OpenMath
- Closely mimic TeX behaviour.
- Lossless
- Extensible, not necessarily in TeX.
- Adaptable.
- . . . and finish DLMF project!

Digital Library of Mathematical Functions

Mimic TEX's Digestive Tract

- Mouth *Tokenizes*
- Gullet *Expands*
- Stomach *Digests* — but Augmented!
- Intestines *Builds* Document Tree
- Postprocessing per application
  - math parsing/analysis, math images,
  - graphics, table rewriting, . . .

Digital Library of Mathematical Functions

Practical Math?

- DO allow author macros
- DO cope with quirky TEX
  `\frac12`, `a^xy` vs. `a^{x}y`
- DON'T pretend to 'understand' all legacy TEX.
  (at least, not without additional info)
- DON'T require completely explicit `\add{a}{b}`
- DO preserve any semantic clues.
- DO encourage markup that reduces ambiguity
- DO allow author/document specific clarification of notations

Digital Library of Mathematical Functions

Math: Middle Road

- Let LaTeXML deal with TeX quirks.
- Acts as structure-preserving Lexer.
- Bulk of math (for us) not *so* bad
- Use infix parser in postprocessing.
- Focus on ambiguities
  - author/document-specific declarations
  - higher-level markup.

Digital Library of Mathematical Functions

Examples: Declarations

With

```
DefSymbol('U','U','FUNCTION');
```

Now `$U(x)$` gives $U(x)$, as before,
And, after parsing

```
<XMApp>
  <XMTok meaning='U' role='FUNCTION'/>
  <XMTok meaning='x' role='ID'/>
</XMApp>
```

instead of

```
<XMApp>
  <XMTok meaning='InvisibleTimes'/>
  <XMTok meaning='U' role='ID'/>
  <XMTok meaning='x' role='ID'/>
</XMApp>
```

Digital Library of Mathematical Functions

Examples: Higher Level Markup

Define a macro such that

$$\texttt{\textbackslash deriv[n]\{f\}\{x\}} \Rightarrow \frac{d^n f}{dx^n}$$

With the declaration

```
DefConstructor('\deriv[]{any}{any}',
  "<XMApp><XMTok meaning='deriv'/>"
  ."  <XMArg>#2</XMArg><XMArg>#3</XMArg>" ...
```

the constructed tree is

```
<XMApp><XMTok meaning='deriv'/>
  <XMArg><XMTok meaning='f'/></XMArg>
  <XMArg><XMTok meaning='x'/></XMArg>
  <XMArg><XMTok meaning='n'/></XMArg>
</XMApp>
```

Examples: Special Functions

With appropriate TEX macrology:

```
\HyperpFq{p}{q} ⇒ ${}_pF_q$
```

Introduce notion of *evaluating a function at*:

```
\HyperpFq{p}{q}@{a}{b}{z} ⇒ ${}_pF_q\left(a;b;z\right)$
```

or (alternative notation)

```
\HyperpFq{p}{q}@@{a}{b}{z} ⇒ ${}_pF_q\left(\begin{matrix}a\\b\end{matrix};z\right)$
```

Palatable notation? Easier to type than

```
\sideset{_{p}}{_{q}}{\mathop{F}}\left({a \atop b};z\r
```

Examples: Special Functions II

Constructing DOM gives

```
<XMApp>
  <XMTok meaning='HyperpFq'/>
  <XMArg><XMTok meaning='p'/></XMArg>
  <XMArg><XMTok meaning='q'/></XMArg>
  <XMArg><XMTok meaning='a'/></XMArg>
  <XMArg><XMTok meaning='b'/></XMArg>
  <XMArg><XMTok meaning='z'/></XMArg>
</XMApp>
```

and parser can treat args individually,

avoiding guesswork.

Digital Library of Mathematical Functions

Examples: Special Functions III

And from there, MathML

```
<m:mmultiscripts>
  <m:mi>F</m:mi>
  <m:mi>q</m:mi>
  <m:none/>
  <m:mprescripts/>
  <m:mi>p</m:mi>
  <m:none/>
</m:mmultiscripts>
<m:mo>&ApplyFunction;</m:mo>
<m:mrow>
  <m:mo>(</m:mo>
  <m:mtable>
    <m:mtr><m:mtd><m:mi>a</m:mi></m:mtd></m:mtr>
    <m:mtr><m:mtd><m:mi>b</m:mi></m:mtd></m:mtr>
  </m:mtable>
```

Digital Library of Mathematical Functions

Problems

- Role of text and spacing in math.
- Overloading of *symbols* (scoping?)
- Palatable LaTeX extensions for math.