Structure Sharing in OpenMath

(c) Michael Kohlhase

Supported by the German Research Council under a Heisenberg grant

http://www.cs.cmu.edu/~kohlhase

Pittsburgh, USA

Saarbrücken, Germany

Universität des Saarlandes

School of Computer Science

Michael Kohlhase

Faculty of Informatics
Wish: Cross-referencing for 
O MATH 

• Status: OMDoc just went ahead (licensed by the OPE N MATH standard) 

Proposal: Extend OM\texttt{E} with 

- \texttt{id} attribute (for OM\texttt{xref} sharing) 
- \texttt{xref} attribute (OM\texttt{xref} (sharing), \texttt{MATHML} \texttt{xref} (semantics)) 
- \texttt{OMEL} with \texttt{xref} empty, \texttt{xref} points to element with same name 

(cleaned up version for \texttt{OMN\texttt{E}}) 

Problems: not based on X\texttt{LINK} yet, semantics differs from \texttt{MATHML} 

Advantages: sharing of (sub)-formulas (simple transformatioin to standard \texttt{OMN\texttt{E}}) 

- Semantics by copying: \texttt{xref} empty, \texttt{xref} points to element with same name 
- \texttt{OM\texttt{E}} with \texttt{xref} \texttt{OM\texttt{E}}, \texttt{OM\texttt{E}}, \texttt{OM\texttt{A}}, \texttt{OM\texttt{E}}, \texttt{MATH} 

(content) 

2016 \textcopyright Michael Kohlhase 

Carnegie Mellon
Structure Sharing with Directed Acyclic Graphs

DAG

Tree

$p$ nodes

$2^p - 1$ nodes

$p$
The same in the OpenMath XML encoding
Summary of the Proposal

Idea:

- Allow structure sharing in the XML encoding by
  - straw-man element OMR (represents target of xlink:href attribute)
  - xlink:href attributes on "fat" OpenMath elements
  - by t<sup>2</sup> attributes on "fat" OpenMath elements

- Pro: OpenMath data model does not change
  - stays finite trees

- Problem: Acyclicity Constraint
  - non-local condition to be verified for validity

Both encodings encode the OpenMath object

© Michael Kohlhase
Carnegie Mellon
Acyclicity Condition

Definition: We say that an element dominates all its children and all elements they dominate.

Problem: Need to traverse the whole document tree to check.

Acyclicity constraint: An element may not dominate itself!
Cyclic data structures can be useful, e.g.

\[
\cdots + \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \ldots
\]

Fun with Cyclic Graphs
ChangestotheDTD

- addthedeclarationfortheOMRelement
  ```xml
  <!ELEMENT OMR EMPTY>
  <!ATTLIST OMR xlink:href CDATA #REQUIRED
  xlink:type CDATA #FIXED 'simple'
  xlink:show CDATA #FIXED 'embed'>
  ```
  so that it reads
  (OMR can only be used inside OMROB)
  (do not make sense on their own)
  (to small, no need)

  but not for elements
  elements OMA, OMBIND, OMATTR, OMI, OMB, OMF,

- addattributelistdeclarations
  ```xml
  <!ATTLIST OMA id ID #IMPLIED>
  ```

- extendtheentitydeclarationfor %omel? so that it reads:
  ```xml
  <!ENTITY % omel "OMS|OMV|OMI|OMB|OMSTR|OMF|
  OMA|OMBIND|OME|OMATTR|OMR">`
A Synopsis of the Landscape of Possible Proposals

<table>
<thead>
<tr>
<th>Character</th>
<th>Pros</th>
<th>Motivation</th>
<th>Legacy?</th>
<th>Reader?</th>
</tr>
</thead>
<tbody>
<tr>
<td>radical</td>
<td>extend algorithms</td>
<td>slightly extend reader</td>
<td>slightly extend reader</td>
<td>complex reader</td>
</tr>
<tr>
<td>innovative</td>
<td>extended DS</td>
<td>allow common DS</td>
<td>save XML space</td>
<td>conservative</td>
</tr>
<tr>
<td>model complex DS</td>
<td>model encoding</td>
<td>model encoding</td>
<td>same data model</td>
<td>innovative</td>
</tr>
<tr>
<td>model complex model</td>
<td>model encoding</td>
<td>model encoding</td>
<td>same data model</td>
<td>radical</td>
</tr>
<tr>
<td>no</td>
<td>yes/need check</td>
<td>yes/need check</td>
<td>yes/need check</td>
<td>Proposal</td>
</tr>
<tr>
<td>CDGS</td>
<td>DAGS in XML</td>
<td>DAGS in XML</td>
<td>DAGS in XML</td>
<td>Proposal</td>
</tr>
</tbody>
</table>