Over 30 years of mathematical excellence

The Numerical Algorithms Group
Combining mathematics and technology for enhanced performance

OpenMath 2: The Next Generation?

Mike Dewar
NAG Ltd
Over 30 years of mathematical excellence

The Numerical Algorithms Group
Combining mathematics and technology for enhanced performance

OpenMath 2: The Next Generation?

Mike Dewar
NAG Ltd
Motivation

- Address specific issues and shortcomings which have arisen out of applications development
- Maintain backwards compatibility with OpenMath 1.1 objects
- As far as possible maintain backwards compatibility with existing OpenMath software
- As far as possible fit in with existing XML/Web Standardisation work
Remit (mainly from Pisa 2002)

- Cleaner separation of object model and encodings
- Make XML encoding a full XML application
- Replace DTD for XML encoding with Schema
- Support for non-OpenMath XML in annotations in XML encoding
- Compatibility with RDF-style tools (e.g. for CDs)
- Cross referencing between objects
- Types and attributions

(Subsequently from MONET)
- Enumerated (aka scripted) variables
Background: Relax NG Schema

- New kind of schema proposed by James Clarke and others
- Normative form is XML
- Also has a BNF-like compact syntax which is human-readable and easy to edit
  - Mapping from compact to XML form not unique
- Highly modular – can have a general schema and restrict attributes, contents of elements etc. later
- Can be converted to XSD, DTD etc. (not necessarily exact correspondence)
- Is being used in W3C for XHTML
Relax NG: Implications for OpenMath

- Using Relax NG as the schema language for OpenMath offers the following potential advantages:
  - Can easily restrict generic schema to particular CDs
  - Can restrict first child of OMBIND or OMERROR
  - Can restrict first element of (key, value) pairs in attributes
  - Can automatically generate XSD, DTD etc.

- To make the most of this we need to enhance the object encoding.
Changes to the Object Model

i. Symbol roles
ii. Semantic attributions
iii. Symbol $cdbase$ property
iv. Indexed Variables
v. *Foreign* objects
vi. Relaxing of name restrictions
The optional *role* attribute on a Symbol

- Restrictions where in an OMA the OMS can appear:
  - *binder*: OMS must be first child of OMBIND
  - *attribution*: OMS must be a “key” in an attribution
  - *semantic-attribution*: OMS must be a “key” in an attribution
  - *error*: OMS must be first child of OMERROR
  - *default*: OMS can appear anywhere else

- Note that the default role is not the same as specifying no role
- Most of this information is already contained in STS
- Although part of the abstract description of a symbol, this information naturally lives in CDs
- Can automatically extract this information from CDs to create restricted schemas
attribute versus semantic-attribute

- Currently an OpenMath application is effectively allowed to ignore or strip attributes
- Attributes now used for things like type information or enumerated variables, which arguably shouldn’t be ignored
- We propose that, given a (key, value) pair in an OMATTR
  - the attribute pair is ignorable if the key has no role or has role attribute
  - the attribute pair is not ignorable if the key has role semantic-attribute
The optional \texttt{cdbase} attribute for symbols

- Currently names are unique within a given CD but there is no way of preventing two CDs having the same name.
- When operating in RDF-like environments, everything needs a unique URI.
- The \texttt{cdbase} attribute is a URI “stub” which can be used to disambiguate two CDs with the same name, or to construct a URI according to the scheme:
  \begin{quote}
  \texttt{cdbase} + “/” + \texttt{cd-name} + “#” + \texttt{symbol-name}
  \end{quote}
  e.g. \texttt{http://www.openmath.org/cds/arith1#plus}
  \texttt{cdbase}
- Note that when no \texttt{cdbase} exists applications may behave as they like.
- Note that although formally an attribute of symbol, encodings may allow \texttt{cdbase} to appear elsewhere.
Enumerated/indexed variables

- Variables now take an optional child which is their cardinality
  - this could be a structure to support enumeration in multiple dimensions
  - this is not intended for general decoration of variables
- This is needed because the second child of an OMBIND is a list of variables, so cannot use an OMS here
- While we could now use semantic-attribute for this purpose it is felt that this is a constructor for an atomic object whereas attributes describe existing objects
Derived and Foreign objects

- **Motivation:**
  - Support arbitrary XML in attributes

- **Generalisation:**
  - Support any “native” format in an attribute, e.g. JPEG, MPEG ...

- **OpenMath 1.1 had two classes of objects: basic and compound**

- **Introduce a third class, derived objects, which can only appear as the second part of an OMATTR pair**

- **Introduce one instance of this class, foreign, which is used to import a non-OpenMath object into an attribution**
Relaxation of restrictions on names

- OpenMath 1.1 names (variables, symbols, CDs) restricted subset of ASCII
- Decision to relax these restrictions to be compatible with XML, Unicode, URI etc.
- Note that original URI spec about to be superseded to allow non-ASCII characters
- Hence OpenMath will try and follow IRI spec
Changes to XML Encoding

- Support full XML syntax
  - removed BNF grammar
  - made DTD non-normative
  - introduce Relax NG Schema (XML version is normative)
- Introduced optional *version* attribute for OMOBJ
- Support for changes to object model
  - *CDBASE* attribute can appear anywhere, OMS inherits in obvious way unless explicitly present
- Support for structure sharing:
  
  ```xml
  <OMA id="foo"> … </OMA>
  <OMR xlink:href="foo"/>
  ```

  - Note that href can be a full URI
Namespaced OMS

- No single popular scheme for dealing with namespaces
  - long discussions at Eindhoven
- Possible to introduce “namespace rich” encodings for OpenMath, but these would be incompatible with existing encoding
- Relax NG + CDBASE give most of the practical advantages we would have obtained
Changes to Binary Encoding

- Support for changes to object model
- Make explicit that all names (symbol, variable, CD etc) are UTF-8 encoded
- Support for sharing, analogous with XML encoding
Changes to CDs

- Re-defined CDs, Signature Files in terms of an abstract data model
  - Existing XML syntax is one possible instantiation of this
- Added new (optional) features:
  - symbol role
  - cdbase
- Deprecated CDUSES
Other Changes

- Many editorial changes
- Replaced DTDs for CDs, Signature Files etc with Relax NG versions
- Revised examples throughout
- Revised bibliography
- Added non-normative appendices for DTDs, XSD Schema, restricted Schemas
Next Steps

- Do we agree with these changes?
- Which is the normative version of the standard?
- Should we change the existing CD collection?
  - +cdbase, -cduses
  - namespaces
- Other proposed changes:
  - “Such That” in OMBIND (Michael Kohlhase)
  - FMPs (James Davenport)
  - …
- OpenMath 2 Primer (Olga Caprotti et al)