OpenMath: Objectives Accomplished

Andreas Strotmann

Universität zu Köln, ZAIK/RRZK

OpenMath Thematic Network Workshop, Helsinki, May 2004

Overview

- **Historical context**
  - „Objectives“ working group
  - Other OpenMath efforts
- OpenMath: Objectives and Achievements
  - Desirable properties
  - Scenarios
  - Architecture
- Outlook and Conclusions

„Objectives“ History

- OpenMath working group
  - Commissioned at 1994 Oxford meeting
  - Objectives working group and mailing list
    - Requirements analysis
  - Members
    - Designated: Richardson, Roelofs, Strotmann, Vorkoetter
    - In addition: van Leeuwen, Abbott; others
  - Proposal January, Endorsed summer 1995
    - ISSAC 1995 poster, journal publication 1998

„Objectives“ (ctd.)

- Contents
  - State of the art
  - Requirements analysis
  - Use cases
    - +Architecture
- Basis for
  - OpenMath Design
  - OpenMath Specification

OpenMath History ctd.

- OpenMath committees (ctd.)
  - Design 1995/1996 (mailing list; report)
  - Communications 1995/1996 (dto., report)
  - Specification
    - 1995/1996 (moved to HTML-Math/MathML)
    - OpenMath draft beta1, summer 1996, Diaz/Gonnet
    - draft beta2 fall 1996 (?), +others
  - EU Consortium 1997 – 2000 / – today
    - OpenMath 1.0 (2000), 1.1 (2002), 2.0 (2004?)

- Ten years of intense efforts

- Have we **accomplished our objectives?**

Overview

- Historical context
  - „Objectives“ working group
  - Other OpenMath efforts
- OpenMath: **Objectives and Achievements**
  - Desirable properties
  - Scenarios
  - Architecture
- Outlook and Conclusions

Measuring Accomplishments

- Against which products do we measure?
  - OpenMath
    - 1.0/1.1/2.0
    - Content Dictionaries
  - MathML
    - Content/Presentation
    - 1.0/1.1/2.0/2.0 2nd revision

Desirable Properties

- -- see ISSAC 1995 poster –

- Expressiveness
- Simplicity
- Flexibility
- Extensibility
- Efficiency

Objective: Expressiveness

- Wide applicability
    - MathML-presentation + OpenMath content CDs: yes
- Many sciences
    - OpenMath: In principle, yes, in practice, not yet
- Any representable mathematical object
    - OpenMath 2: With new shared objects, yes (graph)

Objective: Flexibility

- Many media
  - E-mail
    - Unicode in ASCII: yes – XML: too verbose
  - Copy&paste
    - E.g. MathML-Content Maple <-> IE plugin
  - File storage
  - Inter-process communication
    - XML DOM / MathML DOM
- Accomplished: in principle, yes

Objective: Simplicity

- Easy to implement (system implementors)
  - Via XML libraries: yes, for almost all languages
  - Via OpenMath binary encoding libraries: yes, for a few languages
  - Without XML libraries: not really
  - Semantic-level OpenMath or MathML-Content: fairly complex in practice, but that is unavoidable

– Accomplishment: we‘re close

Objective: Extensibility

- Easy to extend (users and user groups)
  - Content Dictionary maintenance not widely implemented in existing software packages
    - – Do CDs for CA user packages work?
  - Writing of Content Dictionaries fairly easy in principle, but lacks editing tools
    - – But generic XML editing tools work for simple CDs
  - Write-your-own CDs are supported

– Accomplished? Almost!

Objective: Efficiency

- Suitably efficient for
  - Symbolic (highly structured) information
    - XML-encoding: too verbose to be efficient
    - Structure sharing: yes (OpenMath)
    - OpenMath binary encoding: yes (as of version 2)
  - Numerical (lightly structured) data
    - OpenMath binary encoding: good enough
    - XML-encodings: too verbose to be efficient
    - Accomplishment: OpenMath binary is good enough
    - … binary XML is on the horizon…

Objective: Efficiency (ctd.)

- Preserve information
  - Costly / important information
    - OpenMath: via annotations, yes
  - Semantics
    - Within reasonable limits: yes
  - Structure
    - As of OpenMath 2: yes

- Accomplished: yes

Overview

- Historical context
  - „Objectives“ working group
  - Other OpenMath efforts
- OpenMath: **Objectives and Achievements**
  - Desirable properties
  - **Scenarios**
  - Architecture
- Outlook and Conclusions

Scenarios

- Typical scenarios for communicating mathematical information
  - Plug-in scenario
  - E-Mail scenario
  - Typesetter scenario
  - Universal front-end scenario
  - Symbolic computing grid scenario

Judging Scenarios

- Questions for judging accomplishment
  - Is OpenMath/MathML capable of supporting this scenario today?
  - Is OpenMath/MathML the language of choice in this scenario today?
  - Has someone actually realised this scenario with OpenMath/MathML today?
- Consequences
  - If not, why not? Can we change it? How? When?

Plug-in Scenario

- Can be done with MathML+OpenMath
  - Lack of Content Dictionaries problematic
- In the form of copy&paste, has been shown for MathML-Content (OpenMath?)
  - Only language that supports this(?)
- In the form of web-services, say, there is ongoing research

E-Mail Scenario

- It is possible to exchange MathML, OpenMath, CDs via e-mail
- People have presumably done this
- Not yet(?) language of choice for e-mailing formulas
  - Verbosity of XML
  - Lack of built-in math editor for mail clients?
- Dto. for a web page scenario

Typesetter scenario

- Possible only as MathML-Presentation
  - Perhaps with parallel content markup
- Language of choice? Getting there!
  - Implemented in MS Office, OpenOffice…
- Content markup support still very limited
  - Via content to presentation stylesheets
  - Incomplete coverage and localization

Universal Front-End Scenario

- Possible, as MathML + OpenMath
  - But limited support for OpenMath?
  - In practice, need more (e.g. OMdoc?)
- Has anybody done this yet?

Semantic Grid Scenario

- Necessary, but not sufficient, ingredient of semantic grid
- Current research program
  - Practical experience exists in the theorem proving (Calculemus) community
- No method of choice has crystallized
  - However, XML indeed is method of choice
  - MathML is XML method of choice for mathematics

Scenarios Summary

- As a combination, MathML+OpenMath work very well in these scenarios
  - OpenMath alone does not support all
  - MathML alone does not support all
- Some scenarios are still ongoing research
- Still not language of choice everywhere
  - But promising development

Overview

- Historical context
  - „Objectives“ working group
  - Other OpenMath efforts
- OpenMath: **Objectives and Achievements**
  - Desirable properties
  - Scenarios
  - **Architecture**
- Outlook and Conclusions

Architecture

- „Optional“ part of „Objectives“
  - Recommendation
- Language layers and components
  - Relationships between layers or components
  - Proposed as a common ground to integrate existing language definitions

OpenMath Objectives Language Layers

- „Objectives“ define four internal layers
    - + external „application specific representation“
        - Mediated by „Phrasebooks“
  - „OpenMath Object“ layer (multi-branched)
    - + „Lexicon“ component
  - „OpenMath Expression“ layer (single)
  - „OpenMath Data Structures“ layer (single)
  - „OpenMath Encodings“ layer (multiple)

OpenMath Standard Language Layers

- OpenMath 1.0/1.1/2.0 (drafts) define two language layers
        - + external „application specific representation“ („private layer“)
            - Mediated by „Phrasebooks“ (part of „private layer“?)
        - + „OpenMath Content Dictionaries“ (part of „abstract layer“?)
    - „OpenMath Object“ layer („abstract layer“)
    - „Encoded OpenMath Object“ layer („communication layer“)

Translation

- ❖ OpenMath Objectives
  - – Application specific
    - Phrasebooks
  - – ---------------------
  - – OpenMath Object
    - Lexicon
  - – OpenMath Expression
  - – OpenMath Datastructs
  - – OpenMath Encodings

- ❖ OpenMath Standard
  - – Application specific
    - Phrasebooks
  - – -------------------
    - Content Dictionaries
  - – OpenMath Object
  - – Encoded Object

Differences

- Merge „Object“ and „Expression“ layer
  - Distinction based on difference between
    - Structural semantics (universal „categorial semantics“) and
    - Separate (plug-in) lexical semantics
  - Distinction is now implicit, not explicit
- No „data structures“ layer
  - IEEE floats, strings etc. in „Object“ layer instead
  - Structure sharing defined in encodings instead
  - No support for „untagged“ representations
    - Adding these proposed by John Abbott, Nice workshop 2002

Transformations

- „Objectives“ require completeness of transformations between layers
  - – Limits acceptable encodings or semantics
    - OpenMath 1 encodings failed these requirements
  - – No such requirements defined in Standard
    - But OpenMath 2 encodings probably qualify now
    - Standard defines no semantics; criteria N/A
- Accomplished? Yes (OpenMath2)

Accomplishments

- Simplified Architecture
  - Easier to grasp quickly
  - Direct cause for many fruitless discussions
        - (personal opinion!)
      - FPs/BigFloats vs. Int/Bignums
      - Structure sharing
      - „tagless“ representations
      - Role of „roles“
- Accomplished? Good enough!

Overview

- Historical context
  - „Objectives“ working group
  - Other OpenMath efforts
- OpenMath: Objectives and Achievements
  - Desirable properties
  - Scenarios
  - Architecture
- Outlook and Conclusions

Outlook: OpenMath and Datastructures

- Mission accomplished...
- But...
  - – How about representation data structs
    - JAA‘s „untagged“ objects, for example
  - – How about non-symbolic basic objects
  - – How about graph structured object representations
- More powerful XML Schema based data type system should be fitted in between „XML“ and „OpenMath Object“ layers eventually

OpenMath Semantics?

- Disagreement on semantics of Objects
  - Kohlhase: „OpenMath Object as a pure formal data structure / syntax“ (?)
  - Strotmann: „OpenMath Objects have a natural structural semantics“
  - Disagreement is at core of „role“ discussion
    - „Formal syntax only“ -> first define syntax and semantics obeys
    - „Natural semantics“ -> syntax follows semantics

Outlook: Standard OpenMath Semantics

- OpenMath and MathML-Content Semantics
  - Clean, simple, complete, extensible
  - Universal structural semantics (standardizable)
  - Type-system specific lexical semantics (extensible)
  - Combination of these is well-understood (and benign) for a large and interesting class of structural+lexical semantics combinations
  - This is doable! (More research needed though)

Conclusions

OpenMath Objectives Accomplished?

OpenMath Objectives Accomplished.[*][+]

[*] ..., well enough, for now.

[+] ..., with MathML (Presentation and Content) included in „OpenMath“