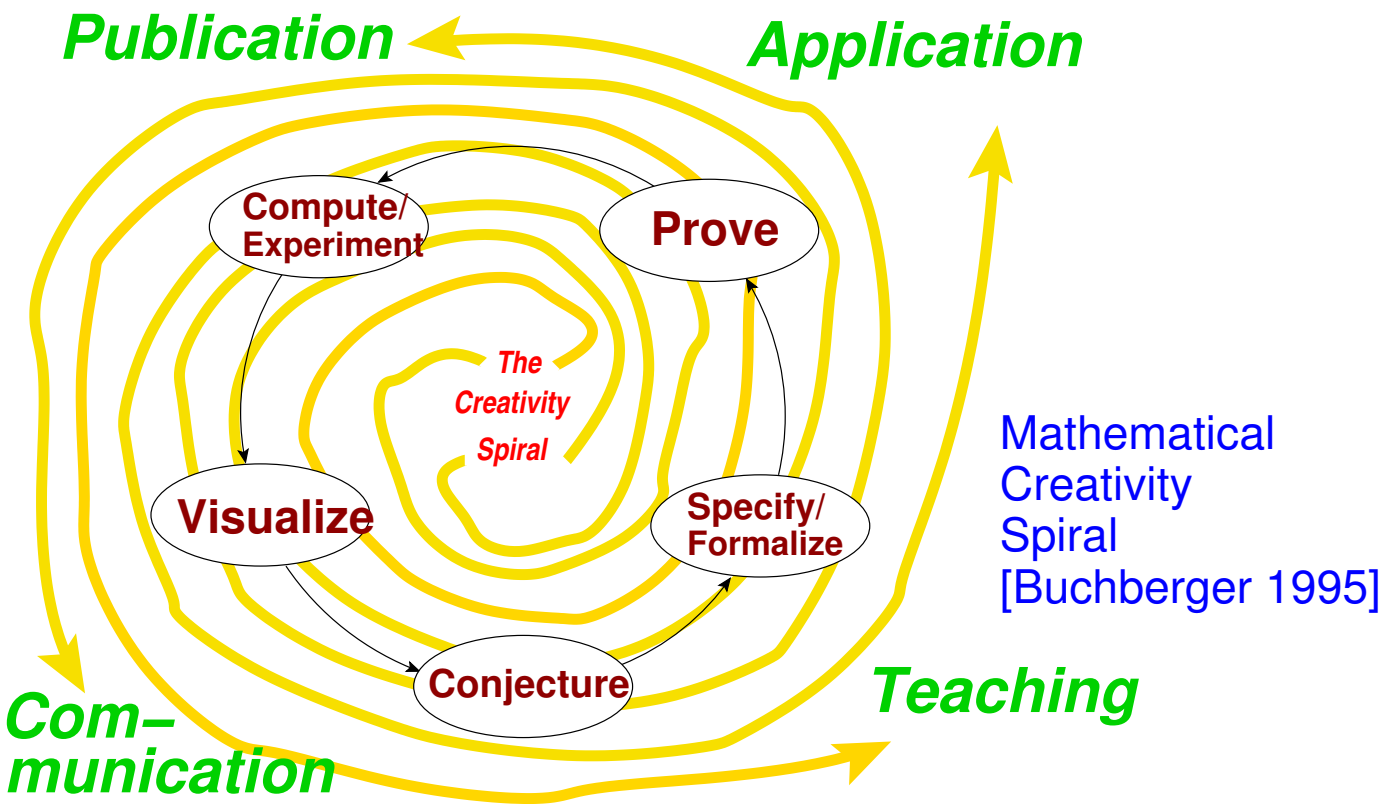
eMath 3.0: Bausteine eines sozialen und semantischen Webs für online-Mathematik

Michael Kohlhase

m.kohlhase@jacobs-university.de
Computer Science
Jacobs University
Bremen, Germany

31. August 2010, Source Talk 2010

The way we do math will change dramatically



- Every step will be supported by mathematical software systems
- Towards an infrastructure for web-based mathematics!

eMath 3.0: The Time is Ripe

- Background:
  - Web 2.0 is the term used for the "social Web" (tagging,blogging,wikis)
  - The "Semantic Web" is a version of the Web, where machines and cooperate
  - Web 3.0 is the term used for the "social Semantic Web".
- We will apply these to eMath ( regular Math may or may not change)

eMath 3.0: The Time is Ripe

- Background:
  - Web 2.0 is the term used for the "social Web" (tagging,blogging,wikis)
  - The "Semantic Web" is a version of the Web, where machines and cooperate
  - Web 3.0 is the term used for the "social Semantic Web".
- We will apply these to eMath ( regular Math may or may not change)
- Recent Improvements of the Math-on-the Web Environment:
  - `MathML3` is Proposed Recommendation (Rec on September 10. 2010?)
  - `MathML` enabled in WebKit (on the road for `Chrome`, `Konquerer`, `Safari`)
  - `MathML` is in HTML5 (without Namespaces though)
  - MathJax has reached 1.0 (Display `MathML` by JavaScript in many Browsers)

eMath 3.0: The Time is Ripe

- Background:
  - Web 2.0 is the term used for the "social Web" (tagging,blogging,wikis)
  - The "Semantic Web" is a version of the Web, where machines and cooperate
  - Web 3.0 is the term used for the "social Semantic Web".
- We will apply these to eMath ( regular Math may or may not change)
- Recent Improvements of the Math-on-the Web Environment:
  - `MathML3` is Proposed Recommendation (Rec on September 10. 2010?)
  - `MathML` enabled in WebKit (on the road for `Chrome`, `Konquerer`, `Safari`)
  - `MathML` is in HTML5 (without Namespaces though)
  - MathJax has reached 1.0 (Display `MathML` by JavaScript in many Browsers)
- Recent Improvements of the Semantic Web:
  - RDF can be embedded into XML via RDFa (linked data export)
  - RDF querying via SPARQL (modulo OWL Ontologies) (semantic search)
  - OMDoc as a mathematical Ontology format (modularity, documentation, full Math)

eMath 3.0: The Time is Ripe

- Background:
  - Web 2.0 is the term used for the "social Web" (tagging,blogging,wikis)
  - The "Semantic Web" is a version of the Web, where machines and cooperate
  - Web 3.0 is the term used for the "social Semantic Web".
- We will apply these to eMath ( regular Math may or may not change)
- Recent Improvements of the Math-on-the Web Environment:
  - `MathML3` is Proposed Recommendation (Rec on September 10. 2010?)
  - `MathML` enabled in WebKit (on the road for `Chrome`, `Konquerer`, `Safari`)
  - `MathML` is in HTML5 (without Namespaces though)
  - MathJax has reached 1.0 (Display `MathML` by JavaScript in many Browsers)
- Recent Improvements of the Semantic Web:
  - RDF can be embedded into XML via RDFa (linked data export)
  - RDF querying via SPARQL (modulo OWL Ontologies) (semantic search)
  - OMDoc as a mathematical Ontology format (modularity, documentation, full Math)
- Overview over the talk:
  - `MathML3` brings more semantics (strict content Math, elementary Math)
  - integrating `MathML`/LaTeX into the Web 2.0
  - A LaTeX-based Semantic Web for Mathematics

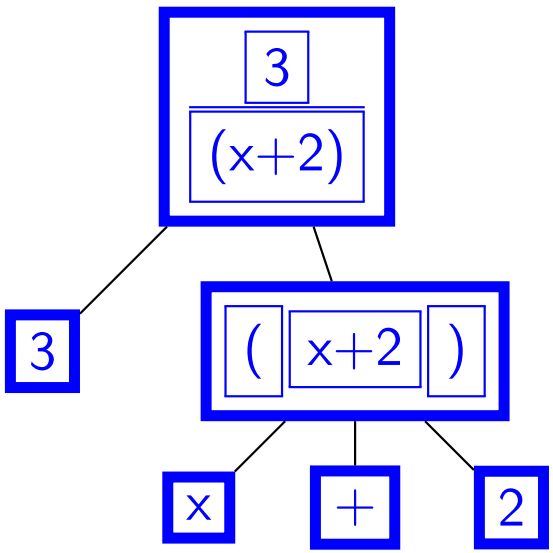
Representation of Formulae as Expression Trees

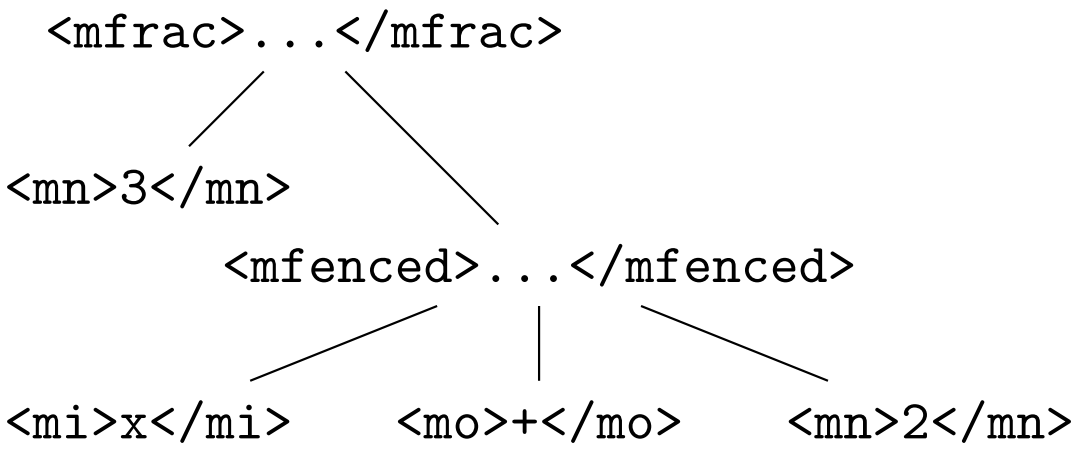
- Mathematical Expressions are build up as expression trees
  - of layout schemata in Presentation-MathML
  - of functional subexpressions in Content-MathML
- Example: $\frac{3}{x+2}$

```
<mfrac>
  <mn>3</mn>
  <mfenced>
     <mi>x</mi>
     <mo>+</mo>
     <mn>2</mn>
   </mfenced>
</mfrac>
```

```
<apply>
  <divide/>
  <cn>3</cn>
  <apply>
    <plus/>
    <ci>x</ci>
    <cn>2</cn>
  </apply>
</apply>
```

Layout Schemata and the MathML Box model





Content Mathml: Expression Trees in Prefix Notation

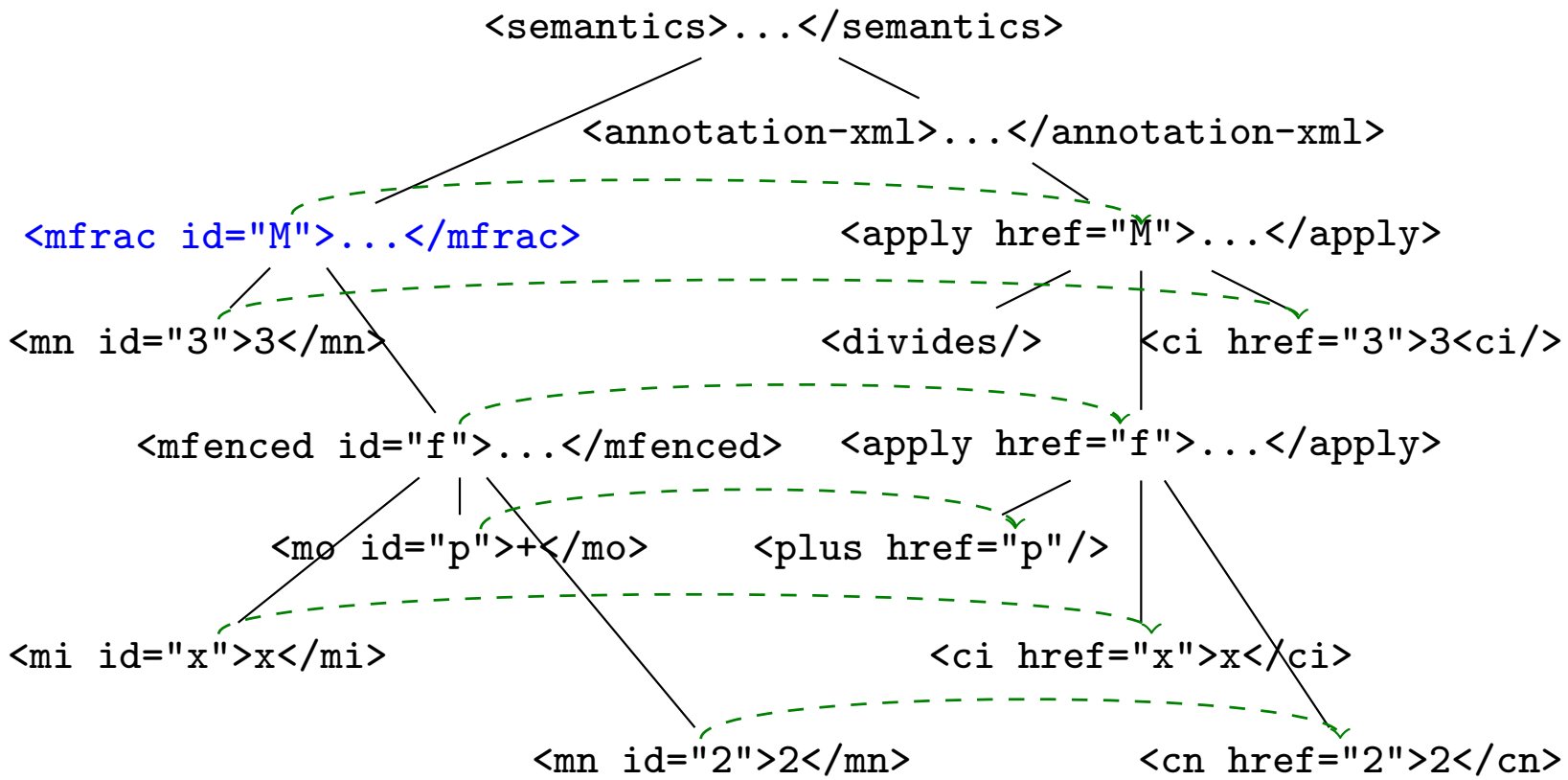
- Prefix Notation saves parentheses (so does postfix, BTW)

$(x-y)/2$	$x-(y/2)$
<apply> <divide/> <apply> <minus/> <ci>x</ci> <ci>y</ci> </apply> <cn>2</cn> </apply>	<apply> <minus/> <ci>x</ci> <apply> <divide/> <ci>y</ci> <cn>2</cn> </apply> </apply>

- Function Application: `<apply> function arg1 ... argn </apply>`
- Operators and Functions: $\sim$ 100 empty elements `<sin/>`, `<plus/>`, `<eq/>`, `<compose/>`,...
- Token elements: `ci`, `cn` (identifiers and numbers)
- Extra Operators: `<csymbol definitionURL="...">...</csymbol>`

Parallel Markup e.g. in MathML

- Combine the presentation and content markup in one tree and crosss-reference



- use e.g. for semantic copy and paste.

(click on presentation, follow link and copy content)

Mixing Presentation and Content MathML

```
<semantics>
 <mrow>
  <mrow><mo>(</mo><mi>a</mi> <mo>+</mo> <mi>b</mi><mo>)</mo></mrow>
  <mo>&InvisibleTimes;</mo>
  <mrow><mo>(</mo><mi>c</mi> <mo>+</mo> <mi>d</mi><mo>)</mo></mrow>
 </mrow>
 <annotation-xml encoding="MathML-Content">
  <apply><times/>
   <apply><plus/><ci>a</ci> <ci>b</ci></apply>
   <apply><plus/><ci>c</ci> <ci>d</ci></apply>
  </apply>
 </annotation-xml>
 <annotation-xml encoding="openmath">
  <OMA><OMS cd="arith1" name="times" />
   <OMA><OMS cd="arith1" name="plus" /><OMV name="a" /><OMV name="b" /></OMA>
   <OMA><OMS cd="arith1" name="plus" /><OMV name="c" /><OMV name="d" /></OMA>
  </OMA>
 </annotation-xml>
</semantics>
```

Web Standards for Formula Markup

language	MathML	OpenMath
by	W3C Math WG	OpenMath society
origin	math for HTML	integration of CAS
coverage	cont+pres; K-14	content; extensible
status	Version 3 (IX 2010)	Version 2 (VI 2004)
activity	propsed rec.	OpenMath3 (compatibility)
Info	http://w3c.org/Math/	http://www.openmath.org/

MathML 3

- Currently under development by the W3C Math Working Group
- Presentation MathML: Incrementally extend functionality
  - manual/automatic linebreaking (for high-quality publishing)
  - arabic math (complex nested R2L/L2R changes)
  - markup for elementary math notations (e.g. long division)
- Content MathML: radical but conservative re-interpretation (give it meaning)
- Strict Content MathML: (OpenMath-isomorphic core)
  regular structure, semantics given by OpenMath objects & CDs
- Pragmatic Content MathML: (MathML2(2e) syntax conserved)
  strike a pragmatic balance between verbosity and formality
- MathML Content Dictionaries with Notation Definitions
- W3C Working Draft at http://www.w3.org/TR/MathML3

C-MathML and OpenMath are equivalent (up to syntax)

- MathML3 introduces a subconcept of Content MathML
  - Strict Content MathML3 $\hat{=}$ OpenMath (semantic core)
  - "Pragmatic" Content MathML3 (semantics by translating into SCMathML)
- **Example 1**

OpenMath	MathML
<OMBIND> <OMS cd="quant1" name="forall" /> <OMBVAR> <OMV name="a" /> </OMBVAR> <OMA> <OMS cd="relation1" name="geq" /> <OMV name="a" /> <OMI>0</OMI> </OMA> </OMBIND>	<bind> <csymbol cd="quant1">forall</csymbol> <bvar> <ci type="nat">a</ci> </bvar> <apply> <csymbol cd="relation1">geq</csymbol> <ci type="nat">a</ci> <cn>0</cn> </apply> </bind>

Elementary Math in MathML3

- "Elementary Math" (worked arithmetics) has specific, CoP-specific presentations
- Problem: Display as math tables is CoP-specific and fragments numbers
- Idea: Introduce specialized markup
- **Example 2** Elementary Math markup for "worked subtraction"

```
<mstack>
  <mscarries crossout='updiagonalstrike'>
    <mn>2</mn><mn>12</mn><mscarry crossout='none'><none/></mscarry>
  </mscarries>
  <mn>2,327</mn>
  <msrow><mo>−</mo><mn>1,156</mn></msrow>
  <msline/>
  <mn>1,171</mn>
</mstack>
```

$$\begin{array}{r} {\scriptstyle 2\,12} \\ 2,3\not{2}\not{7} \\ -1,156 \\ \hline 1,171 \end{array}$$

- Applications:
  - can be presented for different locales (e.g. columns on the base line)
  - we can search for numbers (across table columns)
  - Screen readers like `MathPlayer` can read it out loud,

    see `http://accessiblemath.dessci.com/2010/08/accessible-elementary-math-with-mathplayer-and-mathml-3.html`

Elementary Math in MathML3: Semantics

- Problem: How do we know that this is a subtraction?
- Supposed Answer: by looking at the operator above the line
  <msrow><mo>−</mo><mn>1,156</mn></msrow>
  clearly this is not reliable.
- Idea: use content markup part of parallel markup instead

```
<semantics>
  <mstack>
    <mscarries crossout='updiagonalstrike'>
      <mn>2</mn><mn>12</mn><mscarry crossout='none'><none/></mscarry>
    </mscarries>
    <mn id="n1">2,327</mn>
    <msrow><mo>−</mo><mn id="n2">1,156</mn></msrow>
    <msline/>
    <mn id="n3">1,171</mn>
  </mstack>
  <annotation−xml encoding="MathML−Content">
    <apply>
      <eq/>
      <apply><minus/><mn href="n1">2,327</mn><mn href="n2">1,156</mn></apply>
      <mn href="n3">1,171</mn>
    </apply>
  </annotation−xml>
</semantics>
```

The arXMLiv Project: arχiv to semantic XML

- Idea: Develop a large corpus of knowledge in OMDoc/PhysML
  - to get around the chicken-and-egg problem of MKM
  - corpus-linguistic methods for semantics recovery (linguists interested)
- The Cornell Preprint arχiv: (http://www.arxiv.org) Open access to ca. 600.000 e-prints in Physics, Mathematics, Computer Science and Quantitative Biology.
- The arXMLiv Project: (http://arxmliv.kwarc.info)
  - use Bruce Miller's LaTeXML to transform to XHTML+MathML
  - we have an automated, distributed build system (ca. 1 CPU-year)
  - create ca. 8000 LaTeXML binding files (8 Jacobs students help)
  - use MathWebSearch to index XML version (realistic search corpus)
- More semantic information will enable more added-value services
  - e.g. filter papers by model assumptions (expanding, stationary, or contracting universe)
  - use linguistic techniques to add the necessary semantics

Why reimplement the TEX parser? I

- Problem: The TEX parser can change the tokenizer while at runtime (`\catcode`)
- **Example 3 (Obfuscated TEX)** David Carlisle posted the following, when someone claimed that word counting is simple in TEX/LATEX

```
\let~\catcode~‘76~‘A13~‘F1~‘j00~‘P2jdefA71F~‘7113jdefPALLF
PA’’FwPA;;FPAZZFLaLPA//71F71iPAHHFLPAzzFenPASSFthP;A$$FevP
A@@FfPARR717273F737271P;ADDFRgniPAWW71FPATTFvePA**FstRsamP
AGGFRruoPAqq71.72.F717271PAYY7172F727171PA??Fi*LmPA&&71jfi
Fjfi71PAVVFjbigskipRPWGAUU71727374 75,76Fjpar71727375Djifx
:76jelse&U76jfiPLAKK7172F7117271PAXX71FVLnOSeL71SLRyadR@oL
RrhC?yLRurtKFeLPFovPgaTLtReRomL;PABB71 72,73:Fjif.73.jelse
B73:jfiXF71PU71 72,73:PWs;AMM71F71diPAJJFRdriPAQQFRsreLPAI
I71Fo71dPA!!FRgiePBt’el@ lTLqdrYmu.Q.,Ke;vz vzLqpip.Q.,tz;
;Lql.IrsZ.eap,qn.i. i.eLlMaesLdRcna,;!;h htLqm.MRasZ.ilk,%
s$;z zLqs’.ansZ.Ymi,/sx ;LYegseZRyal,@i;@ TLRlogdLrDsW,@;G
LcYlaDLbJsW,SWXJW ree @rzchLhzsW,;WERcesInW qt.’oL.Rtrul;e
doTsW,Wk;Rri@stW aHAHHFndZPpqar.tridgeLinZpe.LtYer.W,:jbye
```

When formatted by TeX, this leads to the full lyrics of “The twelve days of christmas”. When formattet by LATEXML, it gives

Why reimplement the TEX parser? II

```
<song>
  <verse>
    <line>On the first day of Christmas my true love gave to me</line>
    <line>a partridge in a pear tree.</line>
  </verse>
  <verse>
    <line>On the second day of Christmas my true love gave to me</line>
    <line>two turtle doves</line>
    <line>and a partridge in a pear tree.</line>
  </verse>
  <verse>
    <line>On the third day of Christmas my true love gave to me</line>
    <line>three french hens</line>
    <line>two turtle doves</line>
    <line>and a partridge in a pear tree.</line>
  </verse>
  <verse>
    <line>On the fourth day of Christmas my true love gave to me</line>
    <line>four calling birds</line>
    <line>three french hens</line>
    <line>two turtle doves</line>
    <line>and a partridge in a pear tree.</line>
  </verse>
  ...
```

Why reimplement the TEX parser? III

- But the real reason is: that we can take advantage of the semantics in the LATEX.
- LATEXML does not need to expand macros, we can tell it about XML equivalents.
- **Example 4 (Recovering the Semantics of Proofs)**
  Add the following magic incantation to `amsthm.sty.ltxml` (LATEXML binding)

```
DefEnvironment('{proof}',"<xhtml:div class='proof'>#body</xhtml:div>");
```

The arXMLiv approach: Try to cover most packages and classes in the arχiv
(Jacobs undergrads' intro to research)

Future Plans for arXMLiv

- State: LaTeX-to-XHTML+MathML Format Conversion works (65% success)
- Over the summer: Bump up success rate to 75%, daily downloads, web site, instrumentation,...
- Soon: Integrate user-level quality control (integrate JS feedback into html)
- starting Fall: Extend post-processing by linguistic methods for semantic analysis
  - build semantics blackboard/database for linguistic information (rdf triples)
  - extend build system for arbitrary XML2BB processes
  - invite the linguists over (they leave semantics results in BB)
  - harvest the semantics BB to get OMDoc representations

Current and Possible Applications

- the arxmliv build system `http://arxmliv.kwarc.info`
- the transformation web service `http://tex2xml.kwarc.info`
- LaTeXML daemon to avoid `PERL` and LaTeX startup times (Deyan Ginev)
  - keep LaTeXML alive as a daemon that can process multiple files/fragments (patch memory leaks)
  - a LaTeXML client just passes files/fragments along ($\frac{10}{s}$ to $\frac{100}{s}$)
- embedding/editing LaTeX in web pages `http://tex2xml.kwarc.info/test`
- a MathML version of the arXiv allows vision-impared readers to understand the texts
- generalization search (need to know sentence structure for detecting universal variables)
- semantic search by academic discipline or theory assumption (need discourse structure)
- development of scientific vocabularies (over the past 18 years; drink from the source)

PlanetMath

- **Definition 5** PlanetMath is a wikipedia-like encyclopedia for Mathematics formed by Aaron Krowne in 2001 as an answer to the MathWorld takedown. It is hosted at `http://planetmath.org`.
- PlanetMath contains ca. 8500 encyclopedia articles covering large areas of advanced highschool and undergraduate Mathematics
- PlanetMath is supported by an active user/author community
  - ca. 10 000 unique visitors and 3-5 article updates a day
  - active discussions about articles (ca 30 000 forum posts)
- The underlying software system NOOSphere (based on `PERL`, `LAMP`, JSMath) is showing its age.

Case Study: Re-Engineering PlanetMath

- Observation: PlanetMath is essentially a LaTeX-based forum and wiki (eMath 2.0 system)
- Idea: Re-implement PlanetMath with (PlanetMath(Redux))
  - Vanilla Forum as a well-engineered Web 2.0 platform (ca. 350 000 installations)
  - and `Subversion` for versioning and batch editing
  - LaTeXML for transformation to `MathML`

Case Study: Re-Engineering PlanetMath

- Observation: PlanetMath is essentially a LaTeX-based forum and wiki (eMath 2.0 system)
- Idea: Re-implement PlanetMath with (PlanetMath(Redux))
  - Vanilla Forum as a well-engineered Web 2.0 platform (ca. 350 000 installations)
  - and `Subversion` for versioning and batch editing
  - LaTeXML for transformation to `MathML`
- Case of the eMath 2.0 Case Study: After 6 person weeks of programming: HackUP stage complete (beginning Master's student at Jacobs)
  - factor out new software base Planetary
  - Articles (versioned) and discussions work (edit LaTeX, see XHTML+MathML)
  - page loads 10 times faster than in classic PlanetMath. (PlanetMath(Redux) synchronized via `Subversion`)

Case Study: Re-Engineering PlanetMath

- Observation: PlanetMath is essentially a LaTeX-based forum and wiki (eMath 2.0 system)
- Idea: Re-implement PlanetMath with (PlanetMath(Redux))
  - Vanilla Forum as a well-engineered Web 2.0 platform (ca. 350 000 installations)
  - and `Subversion` for versioning and batch editing
  - LaTeXML for transformation to `MathML`
- Case of the eMath 2.0 Case Study: After 6 person weeks of programming: HackUP stage complete (beginning Master's student at Jacobs)
  - factor out new software base Planetary
  - Articles (versioned) and discussions work (edit LaTeX, see XHTML+MathML)
  - page loads 10 times faster than in classic PlanetMath. (PlanetMath(Redux) synchronized via `Subversion`)
- eMath 3.0 Case Study: Use the Planetary system for semantic markup of my course "General Computer Scinece".

OMDoc in a Nutshell (three levels of modeling)

Formula level: OpenMath/C-MathML

- Objects as logical formulae
- semantics by ref. to theory level

```
<OMA>
 <OMS cd="arith1" name="plus"/>
 <OMS cd="nat" name="zero"/>
 <OMV name="N"/>
</OMA>
```

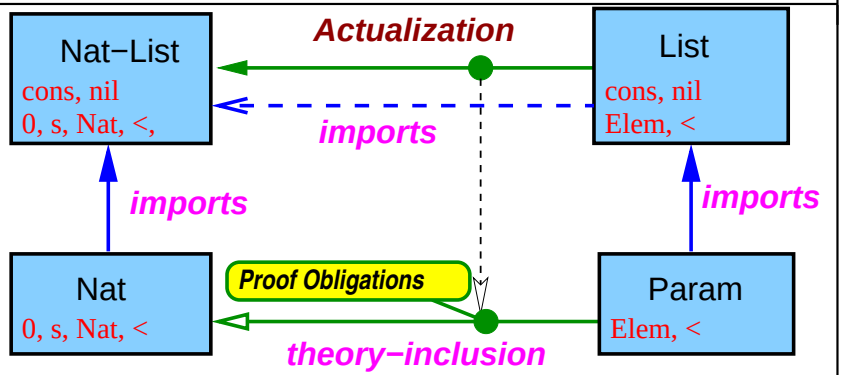
Statement level:

- Definition, Theorem, Proof, Ex.
- semantics explicit forms and refs.

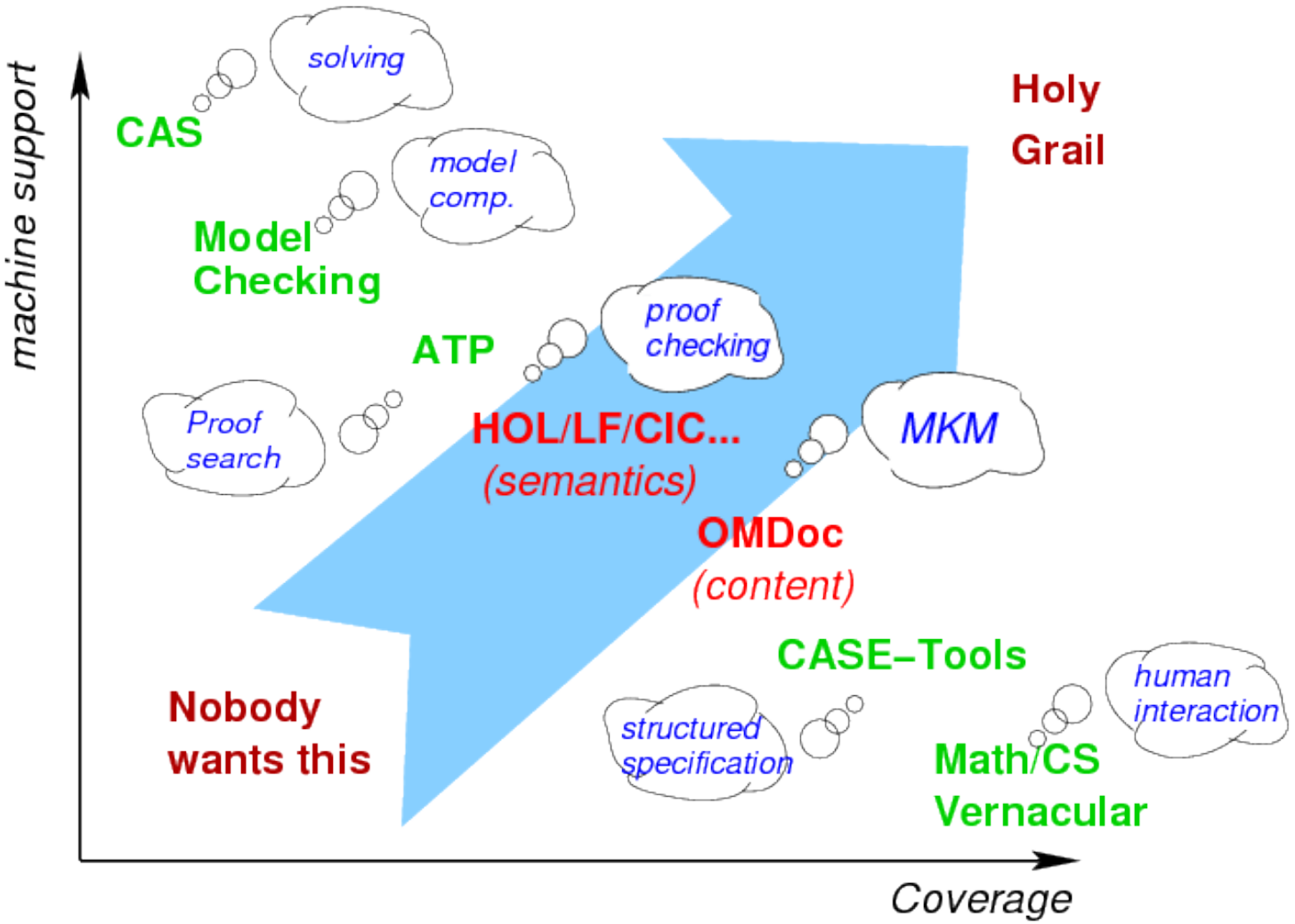
```
<defn for="plus" type="rec">
 <CMP>rec. eq. for plus</CMP>
 <FMP>X + 0 = X</FMP>
 <FMP>X + s(Y) = s(X + Y)</FMP>
</defn>
```

Theory level: Development Graph

- inheritance via symbol-mapping
- theory-inclusion by proof-obligations
- local (one-step) vs. global links



Situating OMDoc: Math Knowledge Management



MKM Formats

- **Definition 6** An MKM format is a content-oriented representation language for mathematics, that makes the structure of the mathematical knowledge in a document explicit enough that machines can operate on it.
- Examples: (so we get a feeling)
  - Document Markup: LaTeX, `DocBook`, `TEI`, OMDoc... (but not TeX)
  - Formula Markup: `Mathematica`, `Maple`, `OpenMath`, Content `MathML` (but not Presentation `MathML`)
  - Theory/Context Markup: `Maya`, `Casl`, OMDoc (but not TeX/LaTeX)
- Goal of this talk: Make TeX/LaTeX into a MKM format on all levels.
  - allow to add explicit structure markup without changing presentation
    in particular, provide infrastructure for formula and theory/context markup.
  - enable translation into traditional MKM formats.
    (solve (part of) the MKM authoring/migration problem)

The MKM Authoring/Migration Problem

- Very interesting systems for Mathematical Knowledge Management (MKM)
- They promise to navigate/index/search/adapt/...large corpora of MK
- Problem: where is the beef?
- Possible sources:
  - libraries from theorem proving- and program verification and computer algebra systems (most of us do that)
  - Write your own in `MathML`/`OpenMath`/OMDoc/... (very tedious)
  - convert from `SGML`/Office engineering documents (difficult to get)
  - adapt from MS PowerPoint documents (works surprisingly well)
  - migrate from existing TEX/LATEX documents (There's the beef)
- TEX/LATEX is a power-user's interface to mathematics!

TEX/LATEX as MKM Format: The Notation/Context Problem

- idiosyncratic notations that are introduced, extended, discarded on the fly

$$\lambda X_\alpha.X =_\alpha \lambda Y_\alpha.Y \hat{=} \mathbf{I}^\alpha$$

  meaning of α depends on context: object type vs. mnemonic vs. type label.
- even "standard notations" depend on the context, e.g. binomial coefficients: $\binom{n}{k}$, ${}_nC^k$, C^n_k, and C^k_n all mean the same thing: $\frac{n!}{k!(n-k)!}$ (cultural context)
- Notation scoping follows complex rules (notations must be introduced)
  - "*We will write $\wp(S)$ for the set of subsets of S*" (for the rest of the doc)
  - "*We use the notation of [BrHa86], with the exception...*". (by reference)
  - "*Let S be a set and $f : S \to S$...*" (scope local in definition)
  - "*where w is the...*" (scope local in preceding formula)
  - Book on group theory in Bourbaki series uses notation [Bou: Algebra]

Observation: Notation scoping is different from the one offered by TEX/LATEX

TEX/LATEX as MKM Format: The Reconstruction Problem

- Mathematical communication relies on the inferential capability of the reader.
- semantically relevant arguments are left out (or ambiguous) to save notational overload (reader must disambiguate or fill in details.)

$$\log_2(x) \text{ vs. } \log(x) \qquad [\![\mathbf{A}]\!]_{\varphi}^{\mathcal{M}} \text{ vs. } [\![\mathbf{A}]\!]$$

- condensed notation: $f(x+1)\pm 2\pi = g(x-1)\mp 2i$ (stands for 2 equations)
- ad hoc extensions: $\#(A \cup B) \leq \#A + \#B$ (exceptions for ∞)
- overt ambiguity: $\sin x/y$ vs. $\frac{\sin x}{y}$ vs. $\sin \frac{x}{y}$ vs. $-1 \leq \sin x/\pi \leq 1$
- size of the gaps varies with the intended readership and the space constraints.
- can be so substantial, that only a few specialists in the field can understand

The sTEX approach

- The reconstruction and the notation/context problem have to be solved to turn or translate TEX/LATEX into a MKM format
- Problem: This is impossible in the general case (AI-hard)
- Idea: Enable the author to make structure explicit and disambiguate meanings
  - use the TEX macro mechanism for this (well established)
  - the author knows the semantics best (at least she understands)
  - the burden is is alleviated by manageability savings (MKM on TEX/LATEX)
- **Definition 7 (sTEX Approach)** Semantic pre-loading of TEX/LATEX documents.
  - Introduce semantic macros: e.g. `\union{a,b,c}` $\rightsquigarrow a \cup b \cup c$
  - Mark up discourse structure: (largely invisible)
    e.g. `\begin{proof}[id=Wiles,for=Fermat]...\end{sproof}`
  - Generate PDF and XML from that (via LATEXML [Miller])

Semantic Analysis of Presentation Math: Symbol Ambiguity

- Atomic Objects: Grounds Terms, Constants, Variables *Type* 1, q , f , ξ, λ
- Complex Objects: Complex Terms, Decorated Symbols
  Type, Assembly, Scope $\Delta^{P_L P_R/2}$, $\Phi_A(p, -1)$
- Operators
  Fixity, Arity, Domain $+, <, \to, \lambda, :, !$
- Delimiters: Brackets, bra-kets, bars, punctuation, spaces
  Scope, Role
- Scripts
  Object constructor, Operator (power,range,predicate) Δ^{∞}, $M_{ij}, \mathbb{N}_{>2}$

Semantic Analysis of Presentation Math: Structural Ambiguity

- Operators
  Precedence, Scope $\sin x/y - z$
- Complex/Vertical operators
  Argument suppression $\lim, \sum_i, \int_0^1$
- Applications
  Glue, Operator suppression $f(a(b-c)), 1\frac{1}{2}$
- Interdependent with symbols
  Disambiguation rollercoaster $C_{\Delta^n,f} = (C_{\Delta,f})^n$

Architecture Outline

Problem	Stage	Implementation	Domain
Correct Modality	Preprocessing	LaMaPUn::Preprocessor::Purify	NLP
Formula Trees	Main	LaMaPUn::SemBlackboard ::FromulaTreeParser	Comp Ling
Underspecification and Ambiguity	Main		Comp Sem Comp Ling
Representation of Phenomena	Postprocessing	XMath, OpenMath Content MathML	MKM

Conclusions & Future Work

- The time is ripe for eMath 3.0
  - We have the necessary building blocks (integration needed)
  - we have first case studies for eMath 3.0 (more to come)
  - General Metaphor: Math Document as an interface to Math Knowledge
- **Conjecture 8** *eMath 3.0 building blocks can be a basis for Math&Elearning*
- Proof: In the pudding: let's do it!

- If this works, then maybe eMath influences pen and pencil Math

Conclusions & Future Work

- The time is ripe for eMath 3.0
  - We have the necessary building blocks (integration needed)
  - we have first case studies for eMath 3.0 (more to come)
  - General Metaphor: Math Document as an interface to Math Knowledge
- **Conjecture 9** *eMath 3.0 building blocks can be a basis for Math&Elearning*
- Proof: In the pudding: let's do it!
- I would like to collaborate with you on
  - adapting/instantiating our tools to your eLearning theory
  - integraging our tools into your eLearning theory (realize it or not, semantics can also help you!)
  - learning from your and your (semantic) needs
- If this works, then maybe eMath influences pen and pencil Math