OMDoc: An Open Markup Format
Mathematical Documents
(A building Block for Web-Based Math)

 Mathematics $\widehat{=}$ Anything Formal 

MICHAEL KOHLHASE

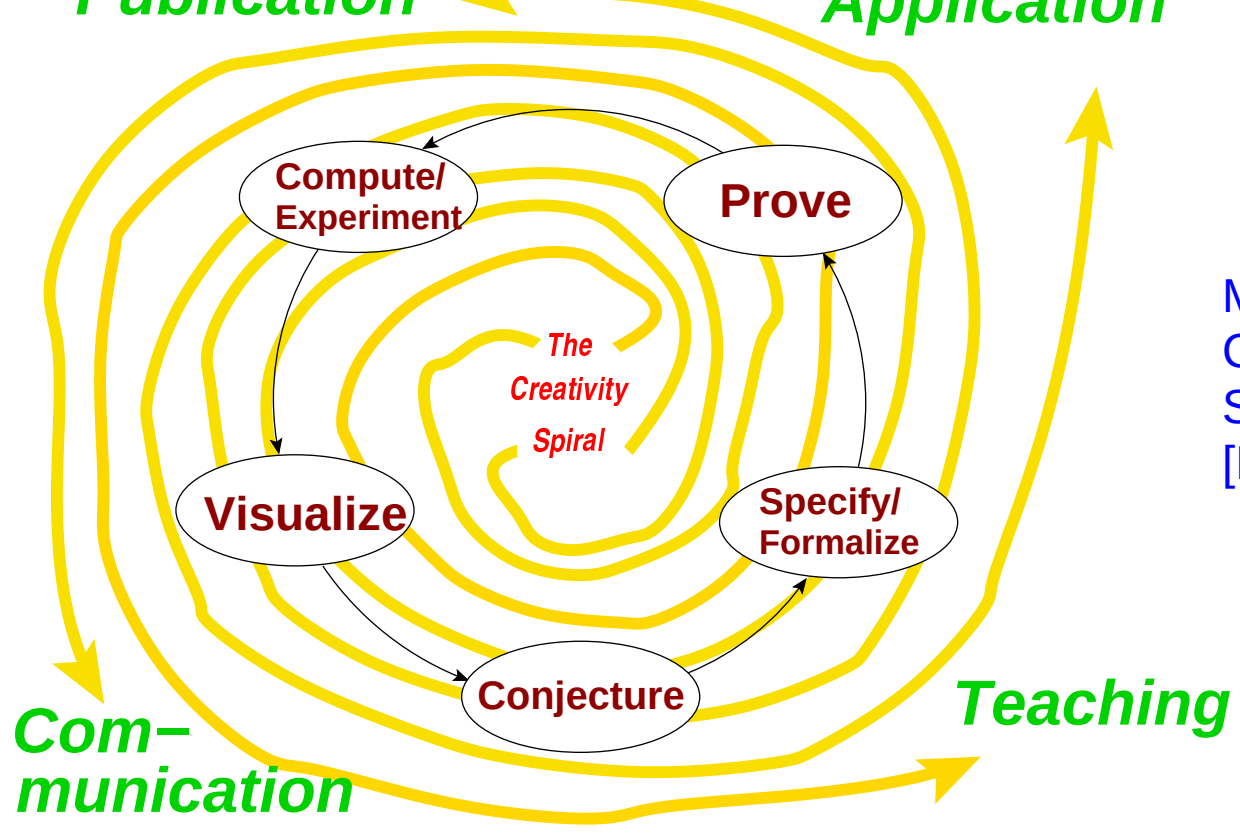
School of Computer Science

Carnegie Mellon University

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

Carnegie Mellon

The way we do math will change dramatically



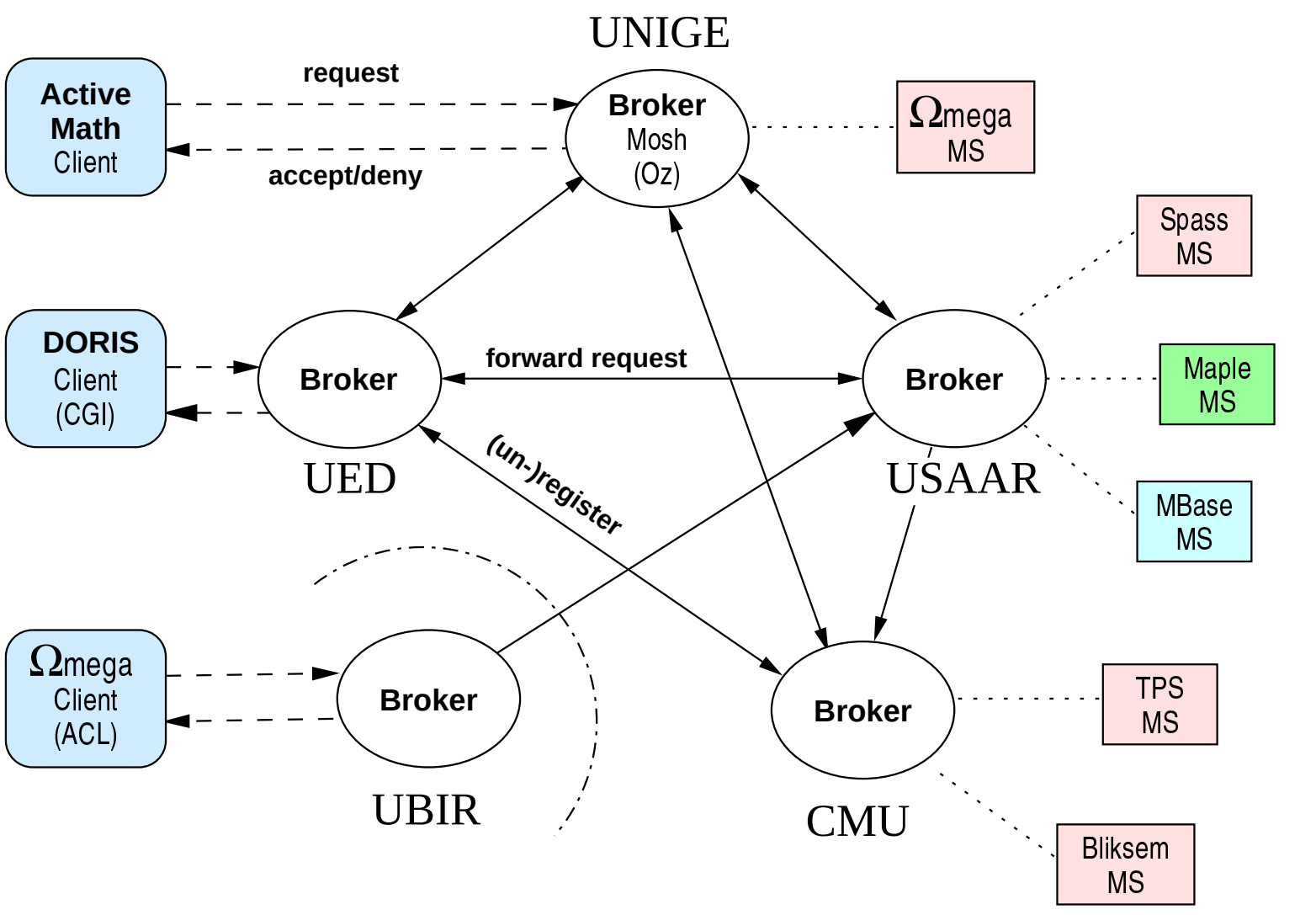
Mathematical Creativity Spiral [Buchberber '95]

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

Towards an Infrastructure for Web-Based Mathematics

- Web-based math requires an open software environment that enables
  - modularization (to keep it manageable)
  - distribution (to use all those machines)
  - inter-operability (to use other peoples systems)
- the MATHWEB approach: (Plug-And-Play mathematics over the web)
  - connect math services by a common math software bus
  - obtain services by encapsulating legacy software or orig. implementation
- Observation: The value and the difficulty lies in the communication
  - we need open standards on every level ($2n$ vs. n^2translations)
  - we need to manage context and semantics (what does it all mean)

The current architecture



Dimensions of Communication among Math Services

Layer	Message	Performative	Content	. . .
Protocol	XML-RPC	KQML	OMDoc	. . .

Level	formulae/objects	doc. elements	theories	service descr.
Protocol	OpenMath/MathML	OMDoc		OMRS. . .

Addressee	content (machine)	presentation (human)
Protocol	OpenMath, C-MathML, Mathematica. . .	Html, PS, PDF, P-MathML

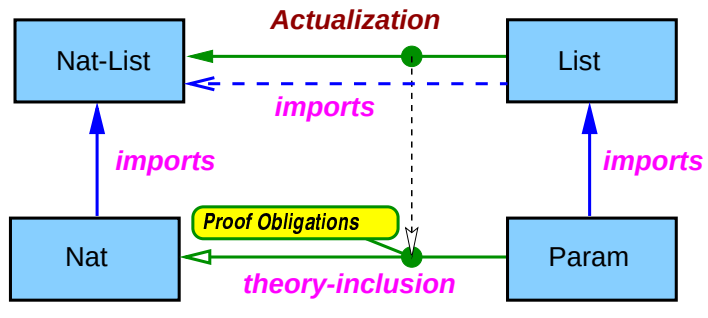
Why mathematics? all the problems of e.g. e-biz in a nutshell

Problem	Efficiency		Safety	
Aspect	translation	context	reference	semantics
Solution	standardize	knowledge base services		standardize

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, Example
• semantics explicit forms and refs.
	`<defn for="plus" type="rec">`
	`  <CMP>rec. eq. for plus</CMP>`
	`  <FMP>X+0 = 0</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	

MBase, a knowledge base of math theories

- This has been attempted before! (Principia Math., Bourbaki,...)
- This time stress the infrastructure aspect (Open Source Model)
  - enable easy and powerful browsing (personalization, MathML)
  - high-level (semantic) search (commutativity: $X(Y,Z) = X(Z,Y)$)
  - distributed Internet support: (e.g. local working KB vs. archive KB)
  - version management & concurrent access (like CVS for cooperation)
  - offer added-value inference services (enlist MathWeb)
  - large-scale structure for navigation & reuse (theory graph, inheritance)
- MBase as a MathWeb component. (not only for human consumption)
  - situated vs. stateless communication of mathematical services

MBase and the semantics of communicated Objects

- Why is this an issue? (preserving the meaning across transport)
  - e.g. the Ricci-Tensor $\mathcal{R}^{ij}$ differs between schools of physicists by a factor of 2!
  - Is this unit in psi or erg? (Remember the Mars orbiter †1999)
- OpenMath/C-MathML approach: Objects as logical formulae
  - sufficient: semantics of constants (that of var, appl, bind is well-known)
  - specify semantics by reference to joint ontology ($\Longrightarrow$ OMDoc)
- MBase as an ontology-server for MathWeb
  - establishes unique reference for objects (distinguished URI)
  - serves knowledge on demand (just-in-time math)
  - offers dynamic communication caches (local context)
  - seamless integration of local and global context
  - Problems: Caching, version management, distributed garbage collection

Added-value services with MATHWEB/MBASE

- cut and paste (cut output from web search engine and paste into CAS)
- automatically proof checking formal argumentations (bridge verification?)
- math explanation (e.g. specialize a proof to a simpler special case).
- semantical search for mathematical concepts (rather than keywords)
- data mining for representation theorems (find unnoticed groups out there)
- classification (given a concrete math structure, is there a general theory?)
- personalized notation (implication as $\rightarrow$ vs. $\supset$, or Ricci as $\frac{1}{1}\mathcal{R}^{ij}$ vs. $2\mathcal{R}^{ij}$)
- user-adapted documents (ActiveMath, Course Capsules)

OpenMath as Representation for math. Objects

- Standard representation for the communication of mathematic objects
- based on Xml (Extensible Markup Language)
- tries to achieve semantic markup. (equv. to content-MathML)
- Objects as (Deliberately uncommitted wrt. object logical)
  - constants (`OMS` symbols) (theory objects)
  - variables (`OMV`) (local objects)
  - applications (`OMA`) (for functions)
  - bindings (`OMBIND`) (λ, quantification)
  - attributions (`OMATTR`) (e.g. types)
- Details at `http://www.openmath.org`

Excursion: XML (for instance OPENMATH)

- XML is language family for the Web
  - tree representation language (begin/end brackets)
  - restrict instances by Doc. Type Def. (DTD) or XMLSchema (Grammar)
  - Presentation markup by style files (XSL: XML Style Language)
- XML= extensible HTML∩ simplified SGML
- logic annotation (markup) instead of presentation!
- many tools available: parsers, compression, data bases, ...
- conceptually: transfer of directed graphs instead of strings.
- details at `http://www.w3c.org`

Example: The OpenMath object $\forall a, b \in \mathbf{R}. a + b = b + a$

```
<OMOBJ>
 <OMBIND>
  <OMS cd="quant" name="forall"/>
  <OMBVAR>
   <OMV name="a"/>
   <OMV name="b"/>
  </OMBVAR>
  <OMA><OMS cd="logic" name="implies"/>
   <OMA><OMS cd="logic" name="and"/>
    <OMA><OMS cd="set" name="in"/><OMV name="a"/><OMS cd="RR" name="real"/></OMA>
    <OMA><OMS cd="set" name="in"/><OMV name="b"/><OMS cd="RR" name="real"/></OMA>
   </OMA>
   <OMA><OMS cd="relation" name="eq"/>
    <OMA><OMS cd="RR" name="plus-real"/><OMV name="a"/><OMV name="b"/></OMA>
    <OMA><OMS cd="RR" name="plus-real"/><OMV name="b"/><OMV name="a"/></OMA>
   </OMA>
  </OMA>
 </OMBIND>
</OMOBJ>
```

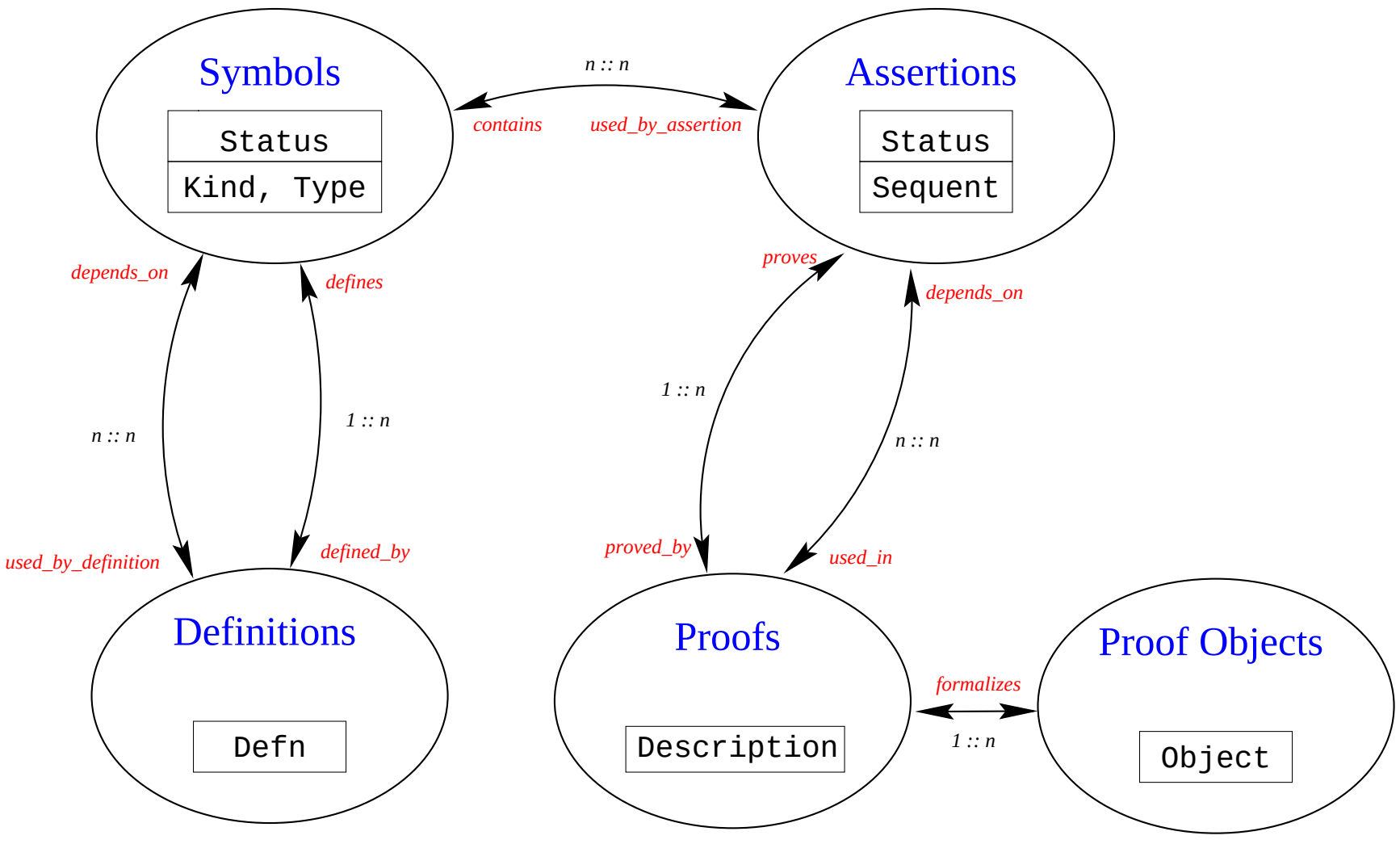
Part of the Arithmetic CD

```
<CD>
  <CDName> arith1 </CDName>
  <CDURL>http://www.nag.co.uk/Projects/openmath/corecd/cd/arith1.ocd </CDURL>
  <CDReviewDate> 2000-09-01 </CDReviewDate>
  <CDStatus> experimental </CDStatus>
  <CDDate> 1999-07-15 </CDDate>
  <CDVersion> 1.02 </CDVersion>
  <CDUses><CDName>alg1</CDName><CDName>integer</CDName>...</CDUses>
  <Description>
    This CD defines symbols for common arithmetic functions.
  </Description>

  <CDDefinition>
    <Name> plus </Name>
    <Description>An nary commutative function plus.</Description>
    <CMP> a + b = b + a </CMP>
    <FMP>∀ a, b. a + b = b + a</FMP>
  </CDDefinition>
</CD>
```

- how can we do better? (look at data!)

An Object-Oriented View of the situation



The Statement Level in OMDoc

- Text elements: `omtext`, `omnote`, `linkage`, `CMP`
- Mathematical elements: (carry the mathematical content)
  - Formulae: `FMP` (contains OpenMath Repns. of logical formulae)
  - Signature: `symbol`, `structure`, `definition`,
  - Properties: `assertion` (`assumption`, `conclusion`) `example`
  - Aux.: `exercise`, `applet`, ...
- Proofs $\widehat{=}$ List of proof steps
  - Steps: `hypothesis`, `derive`, `conclude`
  - in steps: `CMP`, `FMP`, `method`, `premise`, `proof`
- (Meta data: Dublin Core Representation)

Example: an OMDoc Definition

```
<definition id="c6s1p4.d1" for="monoid">
  <CMP>
    A structure (M, *, e) in which (M, *) is a semi-group with unit e
   is called a <with role="definiens">monoid</with>.
  </CMP>
  <FMP>∀M, *, e.s_grp(M, *) ∧  unit(e) ⇒  monoid(M, *, e)</FMP>
</definition>
```

- Make use the co-ocurrence of formal and informal representations by cross-referencing (OMDoc extends OpenMath)

```
<CMP>
  A structure <OMOBJ xref=""/> in which
  <OMOBJ xref=""/>  is a semi-group with
  unit <OMOBJ xref=""/>
  is called a <with role="definiens">monoid</with>.
</CMP>
<FMP>∀M, *, e.s_grp(M, *) ∧ unit(e) ⇒ monoid(M, *, e)</FMP>
```

A Piece of OMDoc Proof

```
<proof id=barshe.2.1.2.proof.a.proof.D2.1>
 <hypothesis discharged-in="D5.2">...</hypothesis>
 ...
 <derive id="barshe.2.1.2.proof.a.proof.D2.1">
  <CMP>By <ref xref="reals.A2"/>  we have z + (a + (−a)) = a + (−a).</CMP>
  <conclusion><FMP>(z + a) + (−a) = z + (a + (−a))</FMP></conclusion>
  <method xref="omega.omdoc#byctx(foralli*@base-calc)"/>
   <parameter>z</parameter>
   <parameter>a</parameter>
   <parameter>−a</parameter>
  </method>
  <premise item="alg-prop-reals.A2"/>
 </derive>
 ...
 <conclude>...</conclude>
</proof>
```

Modular Specification of Presentation Data

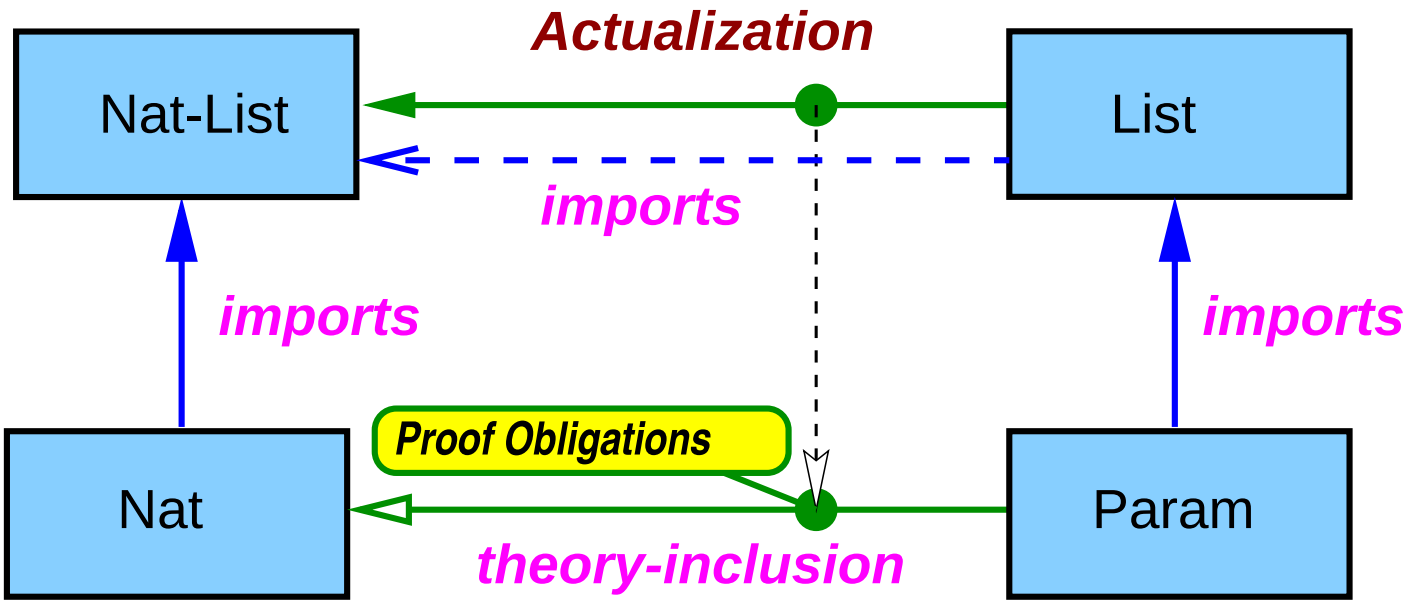
- OMDoc needs presentation style sheets. (raw OMDoc illegible)
- Problem: Documents define new symbols with specialized notations
- Solution: Store presentation information locally (meta-language)
  generate document/collection-specific style sheets

```
<presentation for="binomial" parent="OMA">
 <use format="default" fixity="infix">choose</use>
 <use format="TeX" lbrack="\bigl({" rbrack="}\bigr)">\atop</use>
 <use format="pmml" element="mfrac" attributes="linethickness='0'"/>
</presentation>

<presentation for="power" parent="OMA" fixity="infix"
   crossref-symbol="no" precedence="200" bracket-style="lisp">
 <use format="html" fixity="prefix" element="sup"/>
 <use format="TeX">^</use>
 <use format="pmml" element="msup"/>
</presentation>
```

Theory Management in OMDoc

- Theory management follows Dieter Hutter's development graph model
- Definition and Theorem links (global and local)



- Supports a notion of theory reuse and theory change.
- theories and links represented by special OMDoc elements

A Structured Theory: Lists of Natural Numbers

```
<theory id="Param">
  <symbol id="Elem" type="sort"/>
</theory>

<assertion id="testthm"/>
```

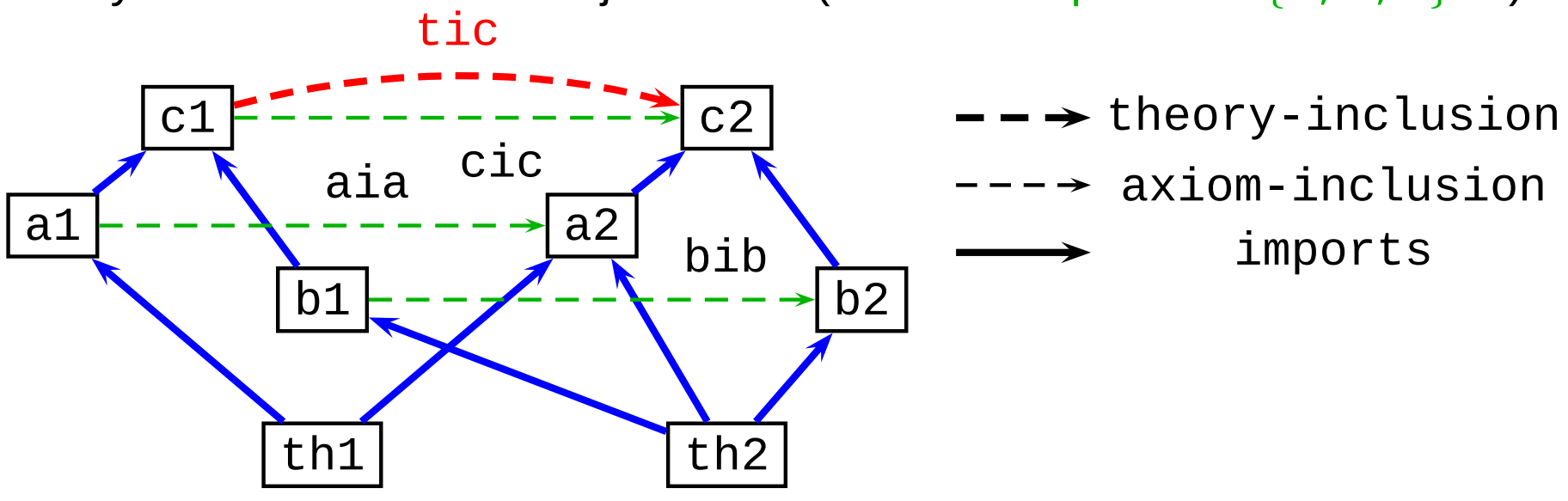
```
<theory id="List">
  <symbol id="List-sort" type="sort"/>
  <symbol id="cons"/>
  <symbol id="nil"/>
</theory>
```

```
<theory id="nat-list.thy">
  <imports id="nat-list.im-nat"     type="global" from="nat-thy"/>
  <imports id="nat-list.im-Element" type="local"  from="List">
    <morphism id="elem-nat"><equation>
        <pattern><OMOBJ><OMS cd="Param" name="Elem"/></OMOBJ></pattern>
        <value><OMOBJ><OMS cd="nat.thy" name="Nat"/></OMOBJ></value>
  </equation></morphism></imports>
  <inclusion item="elem-nat-incl"/>
</theory>

<axiom-inclusion id="elem-nat-incl" from="nat.thy" to="Element" by="testthm">
  <morphism id="elem-nat-incl-morph" base="elem-nat"/>
</axiom-inclusion>
```

Statements about Theories (Theory Inclusions)

- Theory Inclusion: All axioms in the source th. are theorems in the target
- Application: transport of theorems, change of notation. (e.g. Ricci-Tensor)
  - just generate your theory by supplying a morphism that adapts $\mathcal{R}^{ij}$
- Theory inclusions have to be justified (`tic` decomposes to `{a,b,c}ic`)



What math software systems speak OMDoc?

- ΩMEGA, λ*Clam*, INKA, TPS, PVS (theorem provers)
- TRAMP, P.rex (proof explanation)
- MBASE (a mathematical knowledge base)
- presentation systems into LaTeX, HTML, MATHML, (via XSL)
- migration systems from LaTeX, HTML, CASL (⚠ heuristic process)
- transformation to and from OPENMATH CDs
- The ActiveMath math tutor.

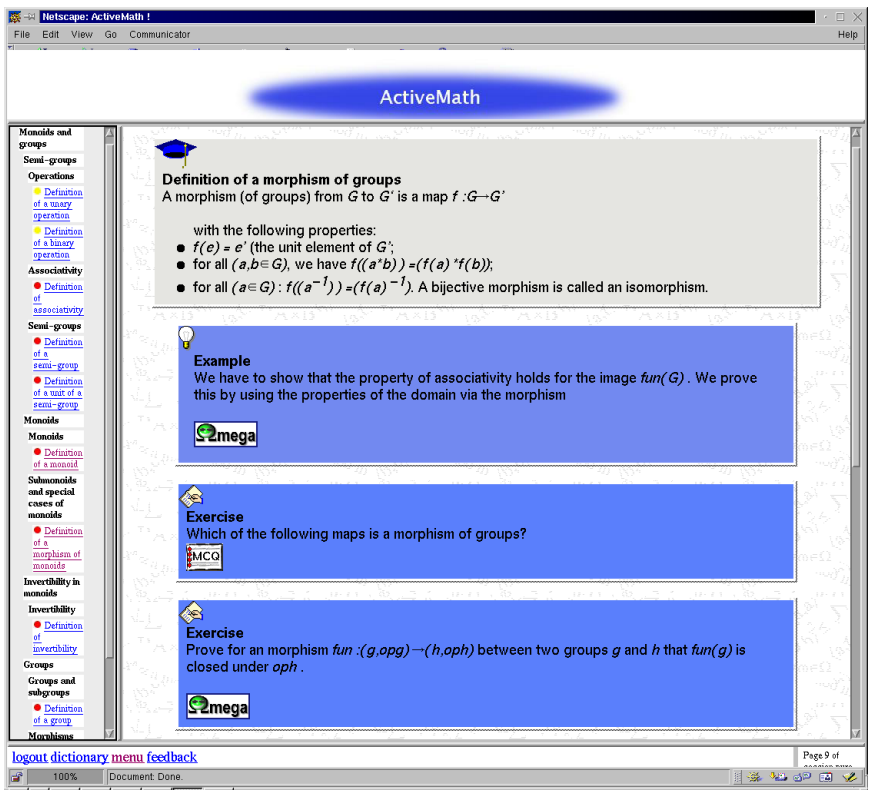
OMDoc/MBase in Web-Based Education

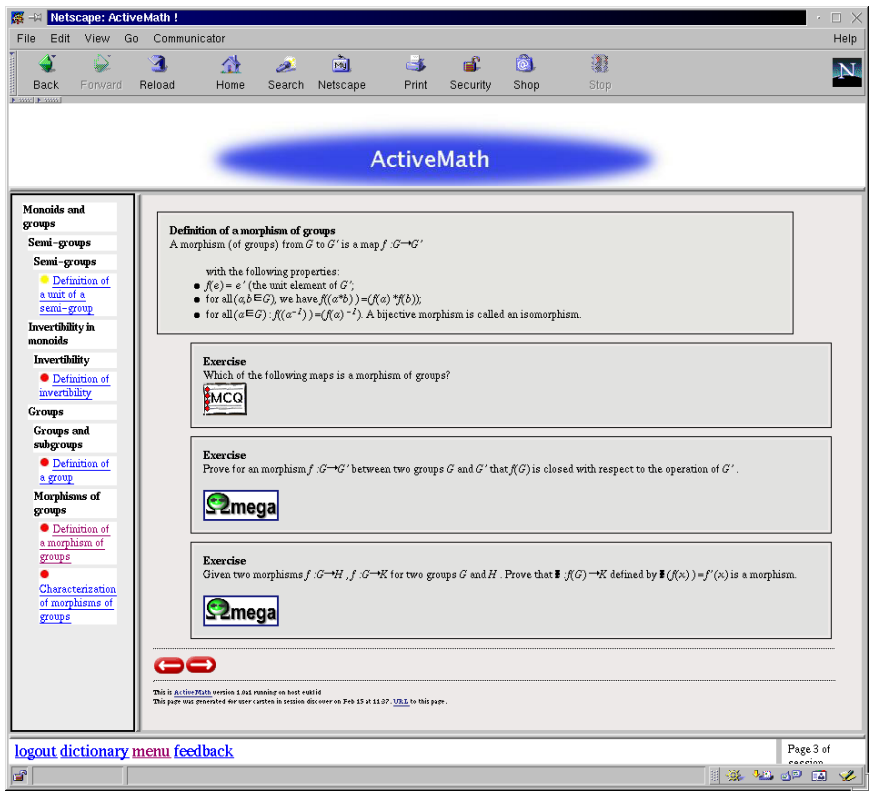
- Formal representation in OMDoc/MBase is a lot of work, but: can be the basis for added-value services that offer
  - Efficient retrieval/searching of content (via MBase)
  - Reuse of content, use of other external systems
- dynamic generation of mathematical documents depending on preferences and abilities of the user
  - Guided tour for a concept (explicit structure)
  - Overview on a certain subject
  - Preparation for exam (examples, problems present)
  - Follow/Review a predefined course (complex document structure)

Experiment: Project ActiveMath, DFKI Saarbrücken

- Content: e.g. Algebra Interactive! ([Cohen et al. 1999])
  - Mountain ($\sim 12MB$) of HTML on a Springer CD.
  - contains (Hyper)Text (Def, Thm, Example, Exercise), Applets, Gaplets
  - Interactive (so-so) but unflexible and unmaintainable ($\Rightarrow$OMDoc)
  - Is used in for undergraduate courses in Eindhoven!
- ACTIVEMATH: personalized, flexible, presentation of knowledge
  - Guided Tours: z.B. transitive hull of the dependency relation
    (Hi, I am Michael, I am interested in the fundamental theorem of algebra)
  - Ext. systems: e.g. the ΩMEGA-System for proof exploration
  - user model: learns about user skills and preferences
    (references object structure)

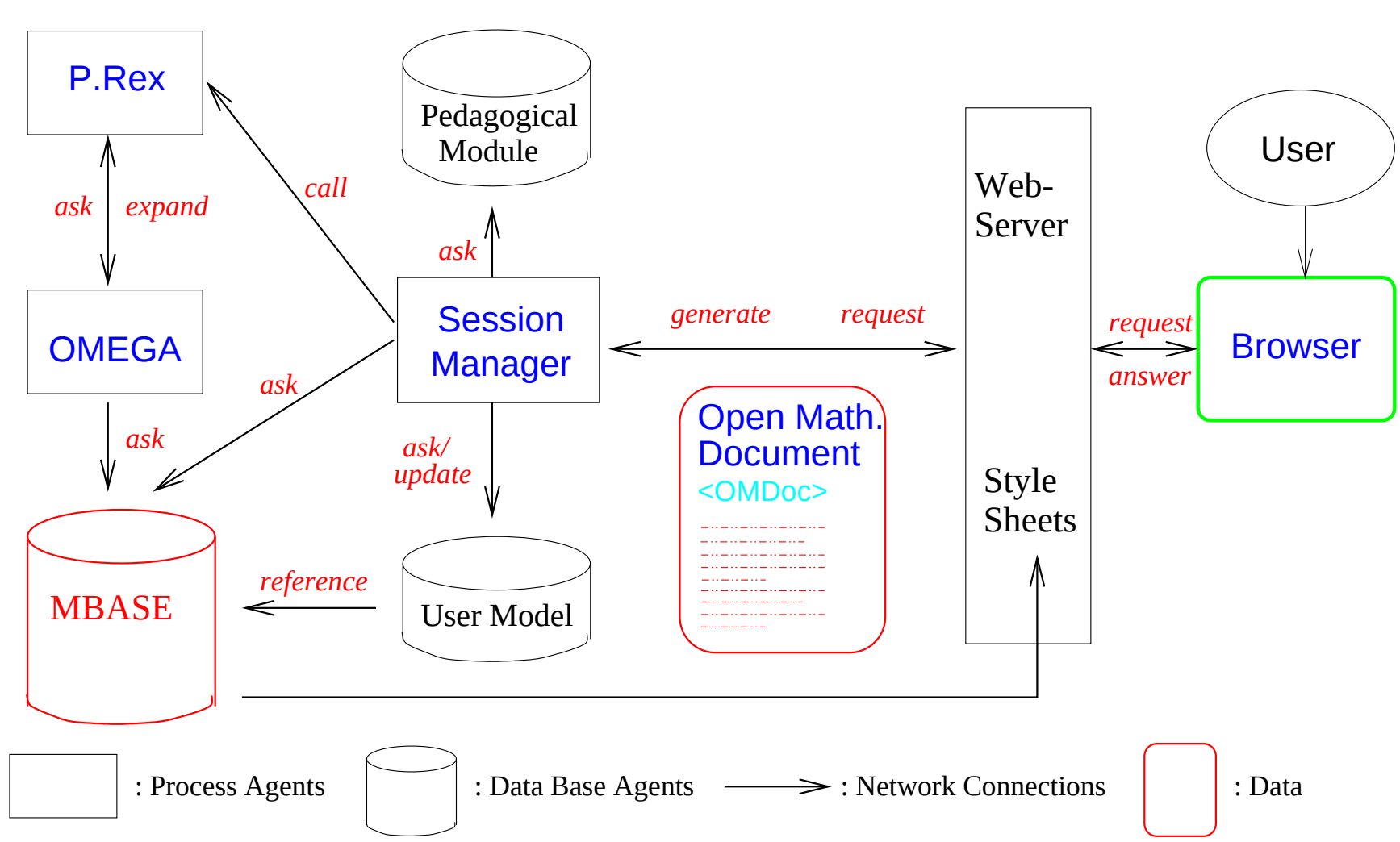
Presentation for Novices and Advanced





- what can be adapted? (via user model)
  - content selection, examples, exercises; availability of systems
  - presentation: level of abstraction, notation, language and modality

ActiveMath: Architecture



Conclusions: how we do math will change dramatically

- A mathematical infrastructure based on communication and knowledge
  - MathWeb as a framework for distributed web-based math services
  - MBase as a knowledge base with added-value services
  - OMDoc standardized communication languages (based on XML)
- gives us the beginning of a mathematical assistant
  - integration of external software systems (make use of the Internet)
  - a common ontology for integration (is zero a natural number?)
  - a universal math repository (data mining, semantical search)
  - bookkeeping in proofs and theories (management of change)
- applications: ΩMEGA and ActiveMath are just the beginning!
  - e.g. Semantic Web Initiative by World Wide Web Consortium (W3C)