The “MathBroker“ Project for Brokering Mathematical Web Services

Wolfgang Schreiner
Wolfgang.Schreiner@risc.uni-linz.ac.at

Research Institute for Symbolic Computation (RISC)
Johannes Kepler University, Linz, Austria
http://www.risc.uni-linz.ac.at

Project Overview

- Long-Term Activity
  - MathBroker I (December 2001 – November 2003).
  - MathBroker II (January 2005 – December 2007).
  - Austrian Science Fund (FWF) contracts P15183 and P17643.
- Coworkers
  - Dr. Olga Caprotti (key researcher in MathBroker I, external coworker in MathBroker II).
  - MSc. Rebhi Baraka (PhD student, MathBroker I + II).
  - MSc. Andreas Duscher (PhD student, MathBroker II).

http://www.risc.uni-linz.ac.at/research/parallel/projects/mathbroker2

Project Goals

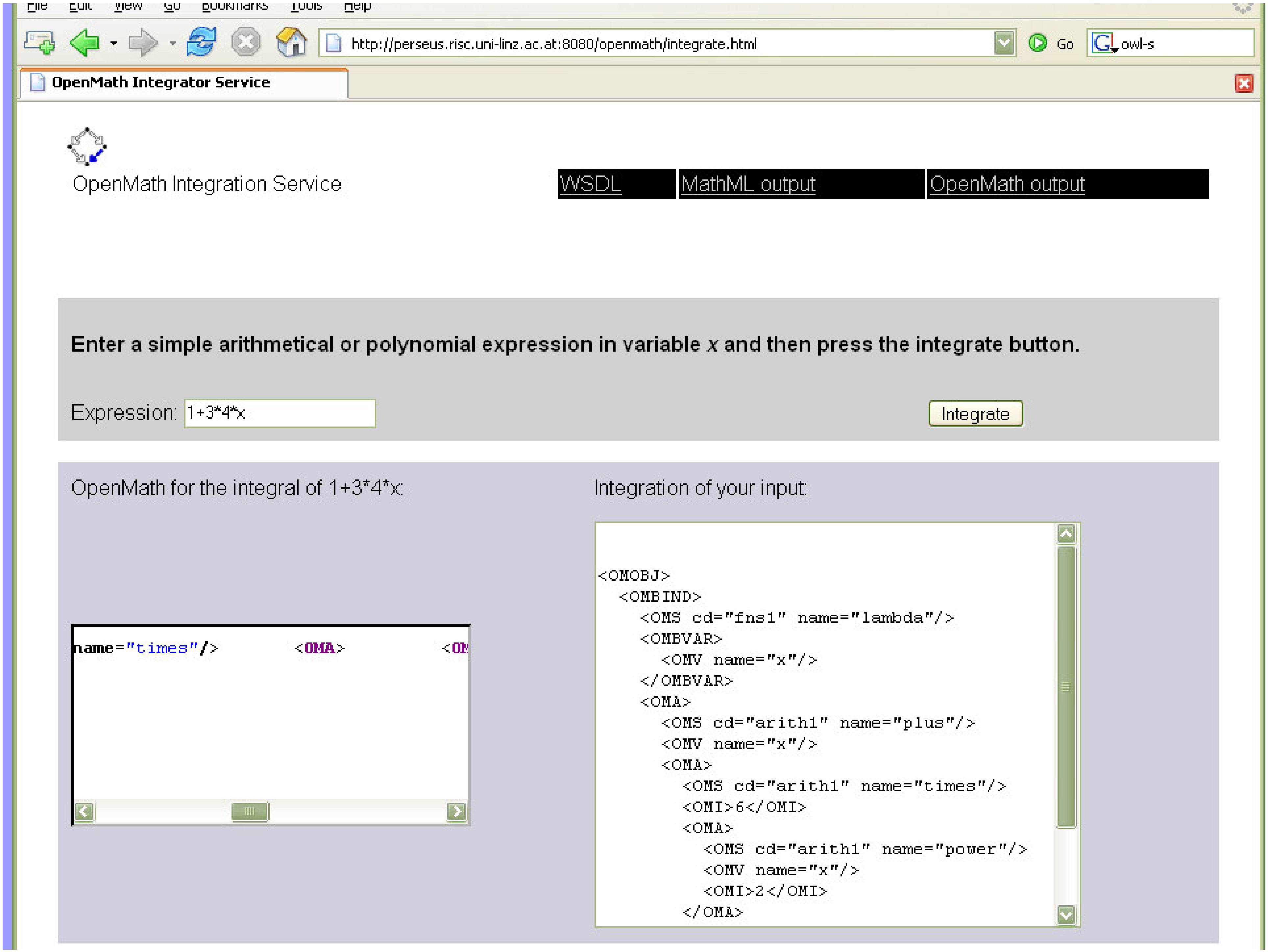
- General
  - Apply technologies from Web Services/Semantic Web to computer mathematics.
  - E.g. computer algebra systems and automated theorem provers as Web Services.
- Vision
  - Software client with a mathematical/logical problem can automatically detect a suitable service, invoke it appropriately for the problem, get the solution.

Project Goals

- Challenges
  - Semantically rich domains.
    - Algebraic formulas, logical propositions, etc.
  - Complex description of service functionality.
    - Not only syntactic interfaces but semantic specifications.
  - Integration of existing symbolic computation software.
    - Mathematica, Maple, GAP, etc.
- Basic technologies
  - OpenMath (OM): semantic representation of mathematical objects in XML encoding.
  - SOAP: communication protocol for Web services.
  - ebXML: basic framework for Web service registries.

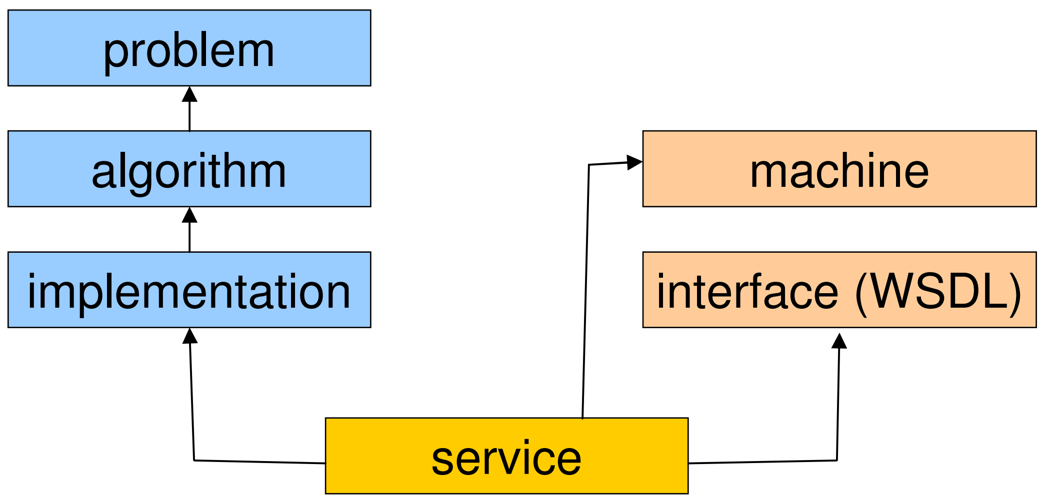
Result (2002): Mathematical Service Execution Framework

- First symbolic computation software framework based on WS standards.
  - AXIS open source implementation of SOAP.
  - OM objects embedded in SOAP messages.
  - WSDL descriptions of services.
  - Software layers: native client $\leftrightarrow$ OM encoding $\leftrightarrow$ SOAP interface $\leftrightarrow$ server $\leftrightarrow$ OM phrasebook $\leftrightarrow$ native mathematical software
  - Examples: Gap, Mathematica sample services.


File Edit View Go Bookmarks Tools Help
http://perseus.risc.uni-linz.ac.at:8080/openmath/integrate.html
Go
owl-s
OpenMath Integrator Service
OpenMath Integration Service
WSDL
MathML output
OpenMath output
Enter a simple arithmetical or polynomial expression in variable x and then press the integrate button.
Expression: 1+3*4*x
Integrate
OpenMath for the integral of 1+3*4*x:
name="times"/> <OMA>
Integration of your input:
<OMOBJ>
<OMBIND>
<OMS cd="fns1" name="lambda"/>
<OMBVAR>
<OMV name="x"/>
</OMBVAR>
<OMA>
<OMS cd="arith1" name="plus"/>
<OMV name="x"/>
<OMA>
<OMS cd="arith1" name="times"/>
<OMI>6</OMI>
<OMA>
<OMS cd="arith1" name="power"/>
<OMV name="x"/>
<OMI>2</OMI>
</OMA>

Result (2003): Mathematical Service Description Language (MSDL)

- Semantic extension of WSDL
  - Co-evolution of MSDL in MathBroker and the European MONET (Mathematics on the Net) project.
  - Hierarchical description structure:

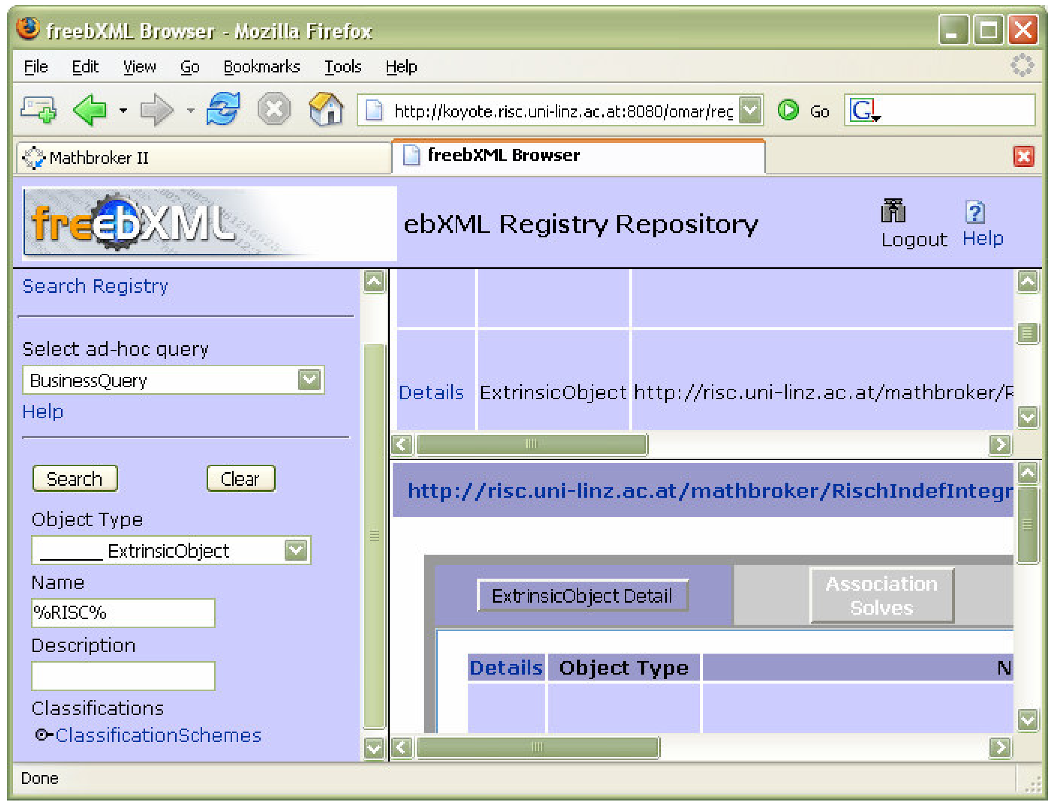


MSDL Example

```
<monet:problem name="indefinite-integration">
    …
    <monet:input name="f">
      <monet:signature>...</monet:signature>
    </monet:input>
    …
    <monet:post-condition>
      <om:OMOBJ>
        <om:OMA>
          <om:OMS cd="relation1" name="eq"/>
          …
      </om:OMOBJ>
    </monet:post-condition>
  </monet:body>
</monet:problem>
```

Result (2003): Mathematical Registry Provider

- Extended ebXML registry provider
  - Registers MSDL descriptions as extensions of ebXML objects.
  - MSDL classifications modeled as ebXML classifications.
  - API for publishing and manipulating MSDL descriptions.



Current Work (2004/2005): Mathemat. Services Query Language (MSQL)

- Example
  - Find all algorithms in GAMS that solve problem *myP* and have deterministc polynomial time complexity.

```
select every algorithm from
  /urn:uuid:56e73807-5d2f-43c8-925a-ec6341b29dcc
where
  //problem[contains(@name, "myP")] and
  //element[@class = "P"]
return algorithm
```

Next Steps (2005/2006): Semantic Queries

- MSQL is still purely syntactic.
  - MSDL contains lot of *semantics.*
- Goal: use semantics for reasoning.
  - Integration of external automatic provers.
- Example:
  - Given a problem P with precondition p and postcondition q, find any service S that solves a problem with precondition p' and q' such that $p \Rightarrow p'$ and $q' \Rightarrow q$ (then S can also solve P).

Next Steps (2005/2006): Service Dialogues

- MSDL describes input/output relation of service.
  - Dynamic aspect of client/server interaction is neglected.
- Service may require a *dialogue* with client.
  - Sequence of messages accepted and returned.
  - Modeling vocabulary and dynamics of dialogue.
  - Basis: OWL-S process ontologies, WS-CDL Web Service Choreography Description Language.
- Goal: client learns language of server.
  - From description, client knows how to interact with a service.

Conclusions

- Web services for computer mathematics.
  - An important but challenging area.
- MathBroker adapts and extends WS technology.
  - WS community puts focus on business applications.
- A first fundament is laid.
  - Mathematical service execution framework, service description language, registry provider, service query language.
- The real challenges still lie ahead.
  - Semantic queries, service dialogues.

http://www.risc.uni-linz.ac.at/research/parallel/projects/mathbroker2