Service Oriented Architectures for Mathematics Assistance Systems

– Motivation, Applications, Technologies –

Christoph E. Benzmüller

Computer Laboratory, Cambridge University

& Saarland University, Saarbrücken

Frankfurt a.M., November, 2006

Overview

- Who am I?
  Own Research in AI, HCI, and Distributed Systems

- What is a Mathematics Assistance System?

- Why Service Oriented Architectures (SOA)
  for Mathematics Assistance Systems?

- Own SOA Application/Technology:
  From MathWeb to MathServe

- How to become a World Champion with MathServe?

Who am I?

Who am I?

- Grown up in Schoden/Saarburg, near Trier and Luxemburg

- Married/living now in Frankfurt-Nordend (just 500m from here)

Who am I?

- Currently working at Cambridge University

- since 2001 head of the OMEGA Group in Saarbrücken:

Dr. Serge Autexier (co-head)
Dr. Chad E. Brown
Dipl.-Inf. Mark Buckley
Dipl.-Inf.Dominik Dietrich
Dr. Armin Fiedler
Dipl.-Inf. Andreas Franke
Dr. Helmut Horacek

Dr. Henri Lesourd
Dipl.-Inf. Marvin Schiller
Dipl.-Math. Ewaryst Schulz
Dipl.-Inf. Frank Theiss
Dipl.-Ling. Dimitra Tsovaltzi
Dipl.-Inf. Marc Wagner
Dipl.-Inf. Jürgen Zimmer

Who am I?

- Other stations in career

Carnegie Mellon University,
Pittsburgh, USA

University of Birmingham,
England

University of Edinburgh,
Scotland

Who am I?

- An enthusiastic middle- and long-distance-runner

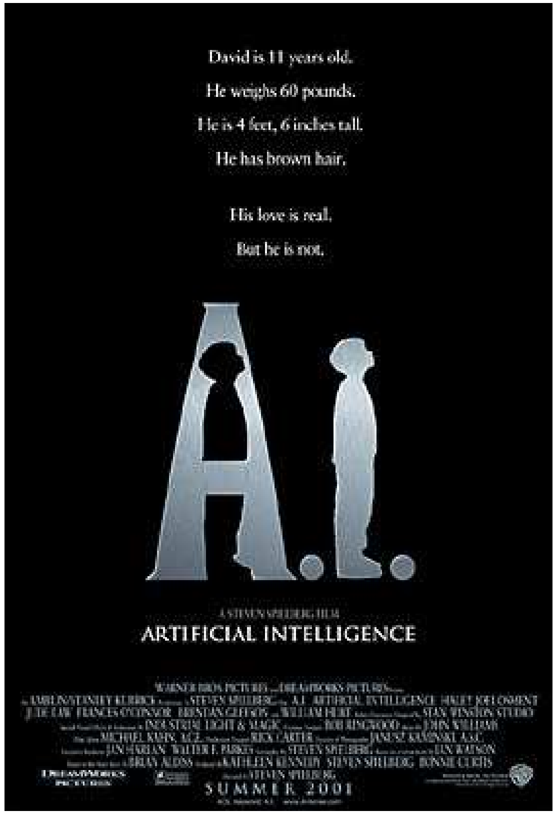
  - German champion 1990 (men's cross-country team)
  - 3rd German championships 1989 over 5000m (Junioren)
  - $>$ 25x Champion of the Rhineland/Rhineland-Palatine
  - Try to beat my personal records:

1000m	2:25min	5000m	14:13min
1500m	3:49min	10000m	30:04min

- So why am I not at the Olympic Games?

Who am I?

- By accident I came into contact with AI and Computer Science



Jörg Siekmann cited Marvin Minsky: “The brain happens to be a meat machine”

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

Research Interest in AI

Can machines think?

Research Interest in AI

Can machines think?

At the end of the century, the use of words and general educated opinion will have changed so much that one will be able to speak of “machines thinking” without expecting to be contradicted.

Alan Turing, 1950

Research Interest in AI

Can machines think?

At the end of the century, the use of words and general educated opinion will have changed so much that one will be able to speak of "machines thinking" without expecting to be contradicted.

Alan Turing, 1950

Can machines play chess? (1997)

Research Interest in AI

Can machines think?

At the end of the century, the use of words and general educated opinion will have changed so much that one will be able to speak of "machines thinking" without expecting to be contradicted.

Alan Turing, 1950

And how about mathematics?
(Can we built intelligent
Mathematics Assistant Systems?)

Can machines play chess? (1997)

Distributed Systems

Distributed System / Distributed Computing: Own Research

Development and application of

- Agent oriented Architectures
- Service oriented Architectures

for designing and realizing intelligent, distributed, robust, scalable, and open

Mathematical Assistance Systems

in the emerging

Mathematical Semantic Web

Human Computer Interaction

Human Computer Interaction: Own Research

- Multi-modal user interaction for mathematics assistence systems:
  - formal logic
  - maths formulas
  - natural language
  - graphics
- E-Tutoring of mathematics
- (psychological) experiments

What is a Mathematics Assistance System?

Overall Scientific Goal

Mathematics Assistance System(s)

Overall Scientific Goal

Mathematics Assistance System(s)

- Computing

Overall Scientific Goal

Mathematics Assistance System(s)

- Computing
- Proving

Overall Scientific Goal

Mathematics Assistance System(s)

- Computing
- Proving
- Exploring/Inventing

Overall Scientific Goal

Mathematics Assistance System(s)

- Computing
- Proving
- Exploring/Inventing
- Illustrating/Publishing

Overall Scientific Goal

Mathematics Assistance System(s)

- Computing
- Proving
- Exploring/Inventing
- Illustrating/Publishing
- Structuring/Organizing

Mathematics Assistance System(s)

- Computing
- Proving
- Exploring/Inventing
- Illustrating/Publishing
- Structuring/Organizing
- Explaining/Teaching

Overall Scientific Goal

Mathematics Assistance System(s)

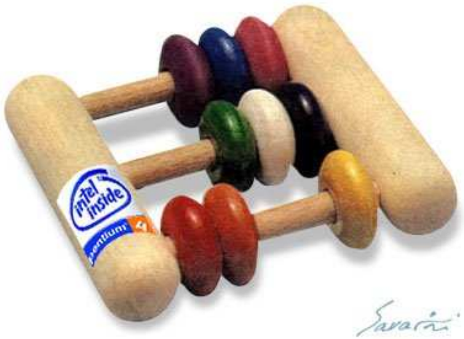
- Computing
- Proving
- Exploring/Inventing
- Illustrating/Publishing
- Structuring/Organizing
- Explaining/Teaching
- . . .

Overall Scientific Goal

Mathematics Assistance System(s)

- Computing
- Proving
- Exploring/Inventing
- Illustrating/Publishing
- Structuring/Organizing
- Explaining/Teaching
- . . .

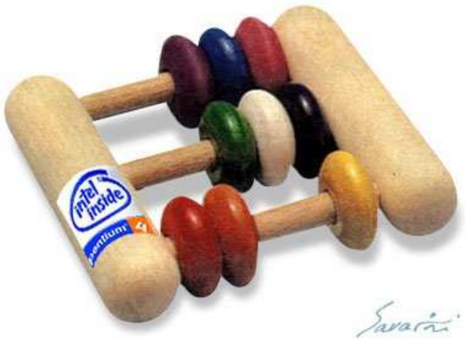
Computing Environments



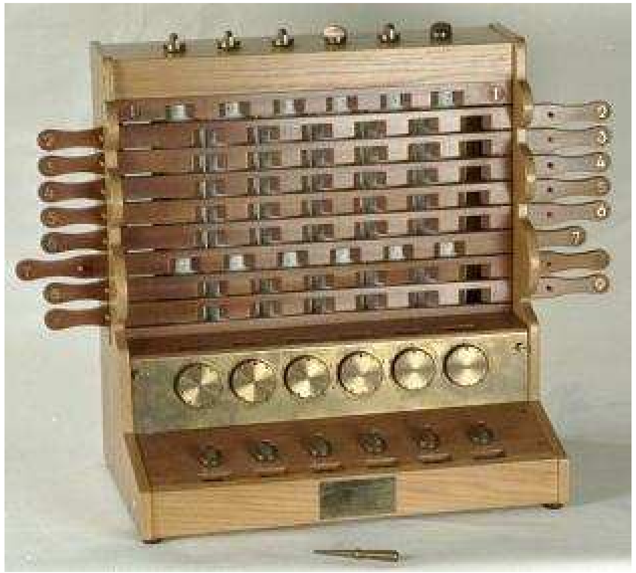
Abacus

(˜500 a.d.)

Computing Environments

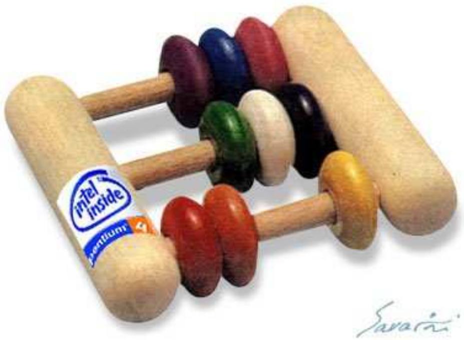


Abacus
(˜500 a.d.)

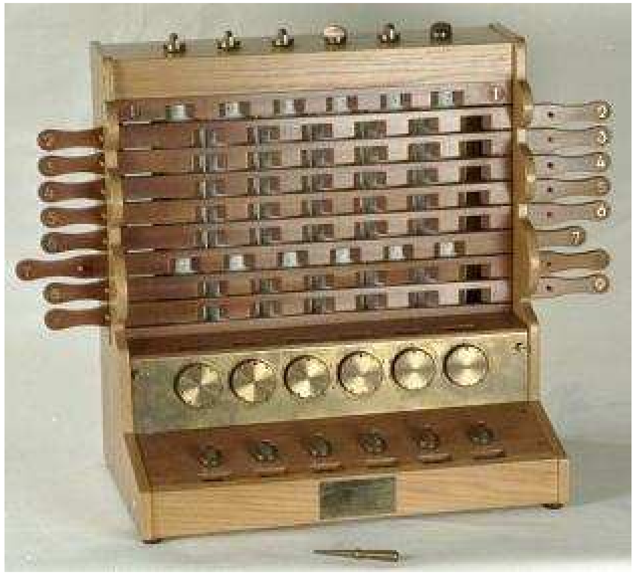


Wilhelm Schickard's Mechanical
Calculator (1592 - 1635)

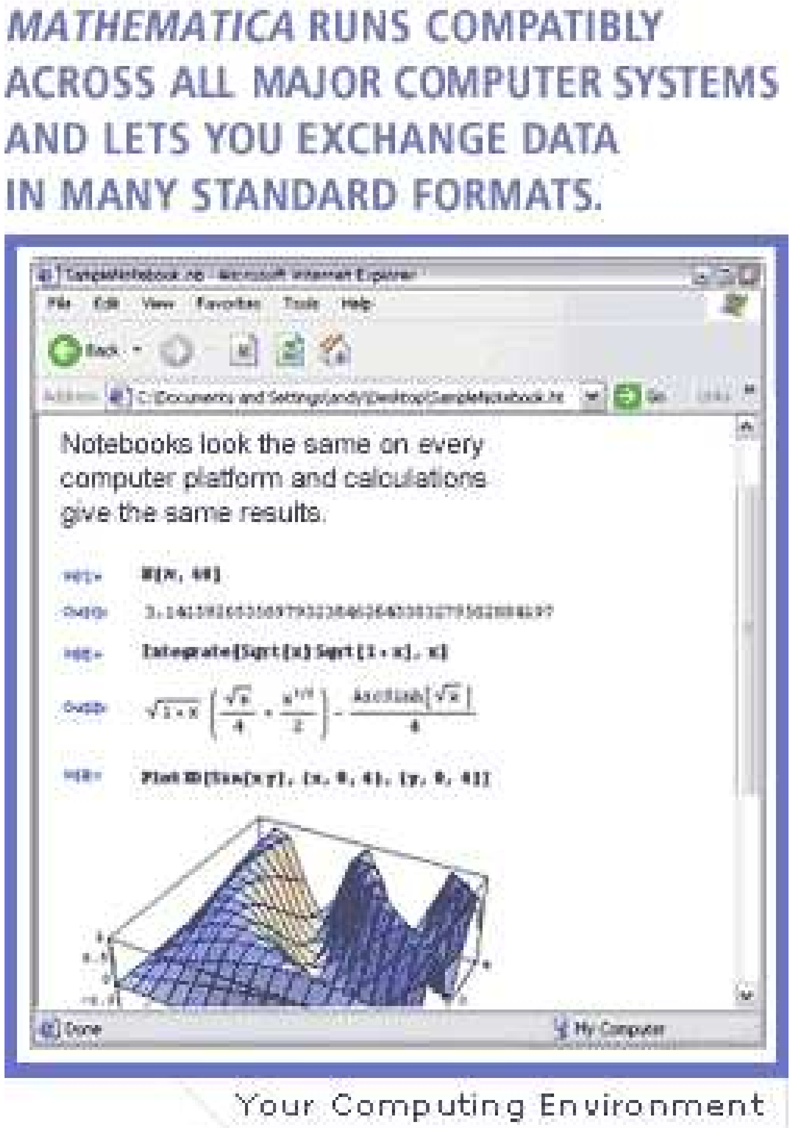
Computing Environments



Abacus
(~500 a.d.)



Wilhelm Schickard's Mechanical Calculator (1592 - 1635)



Modern Computer Algebra System

Theorem Provers: Foundations

Frege, Russel, Hilbert Predicate Calculus and Type Theory

$$\forall x, y, z.(x + (y + z)) = ((x + y) + z)$$

$$\neg \exists f_{n \longrightarrow (n \longrightarrow o)} \forall m_{n \longrightarrow o} \exists x_n . f(x) = m$$

Gentzen Natural Deduction Calculus (Gentzen 1935)

ND Rules (ex.)

$$\dfrac{\mathbf{A} \Rightarrow \mathbf{B} \quad \mathbf{A}}{\mathbf{B}}\ \mathsf{mp}$$

$$\dfrac{\mathbf{A} \quad \mathbf{B}}{\mathbf{A} \wedge \mathbf{B}}\ \wedge \mathsf{I} \qquad \dfrac{\begin{array}{c}[\mathbf{A}]_1 \\ \vdots \\ \mathbf{B}\end{array}}{\mathbf{A} \Rightarrow \mathbf{B}}\ \Rightarrow \mathsf{I}^1$$

$$\dfrac{\mathbf{A} \wedge \mathbf{B}}{\mathbf{A}}\ \wedge \mathsf{E}_l$$

$$\dfrac{\mathbf{A} \wedge \mathbf{B}}{\mathbf{B}}\ \wedge \mathsf{E}_r$$

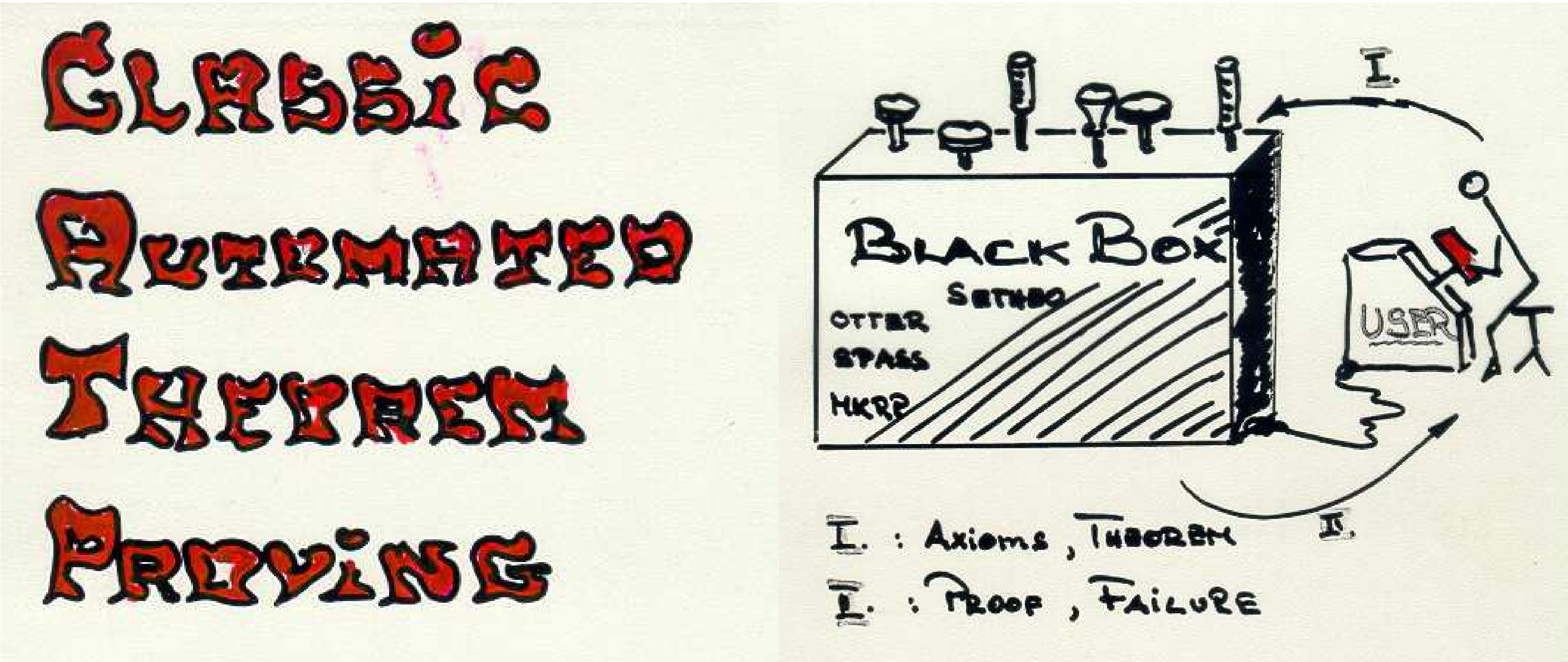
. . . usw. . . .

ND Proof for $(\mathbf{A} \wedge \mathbf{B}) \Rightarrow (\mathbf{B} \wedge (\mathbf{C} \vee \mathbf{A}))$

$$\dfrac{\dfrac{\dfrac{[\mathbf{A} \wedge \mathbf{B}]_1}{\mathbf{B}}\ \wedge \mathsf{E}_r \qquad \dfrac{\dfrac{[\mathbf{A} \wedge \mathbf{B}]_1}{\mathbf{A}}\ \wedge \mathsf{E}_l}{\mathbf{C} \vee \mathbf{A}}\ \vee \mathsf{I}_r}{\mathbf{B} \wedge (\mathbf{C} \vee \mathbf{A})}\ \wedge \mathsf{I}}{(\mathbf{A} \wedge \mathbf{B}) \Rightarrow (\mathbf{B} \wedge (\mathbf{C} \vee \mathbf{A}))}\ \Rightarrow \mathsf{I}^1$$

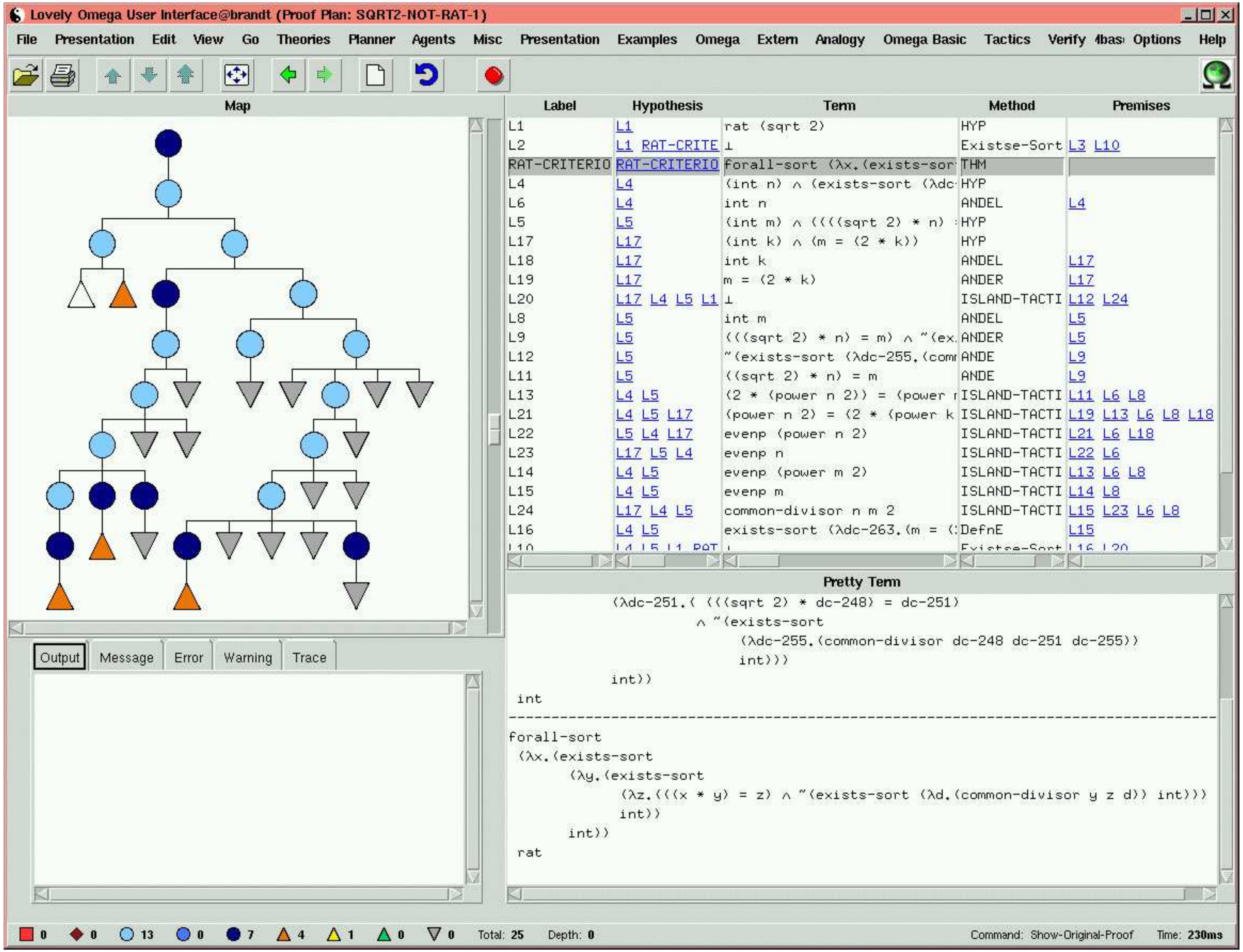
Theorem Provers: Automation

Robinson (1965) Resolution Calculus for Black-Box Automation



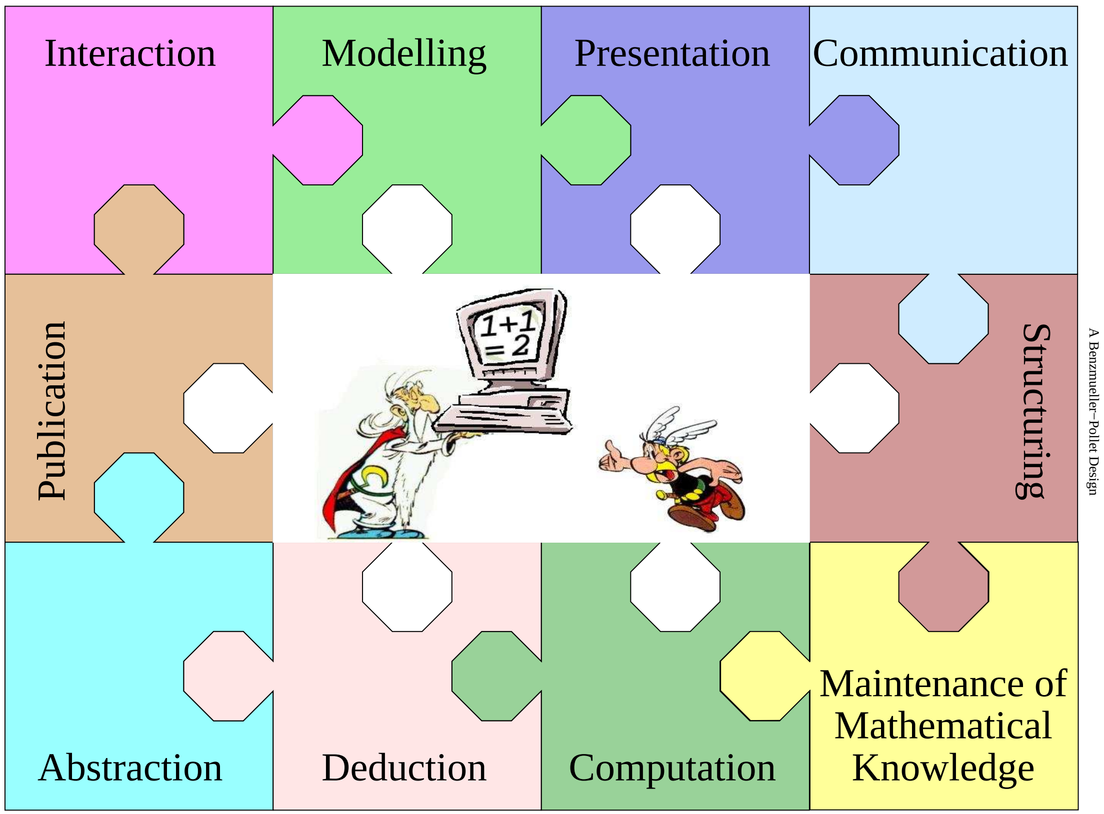
Graphics: Jörg Siekmann

Theorem Provers: Interaction



OMEGA Project: Overall Goal

Integrated Mathematical Assistant Environment vs. 'Pen-and-Paper' Mathematics



Which System Architecture?

Challenging Observation I

- Mathematics assistance systems share many characteristics with large AI systems in general
- Subsystems to be integrated are
  - intelligent systems themselves
  - developed over years
  - commercial and non-commercial
  - use different programming languages, architectures, platforms, etc.
  - steadiliy changing and improving

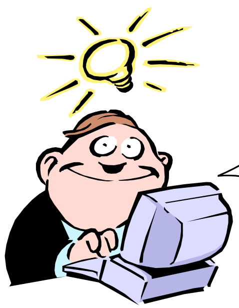
Challenging Observation II

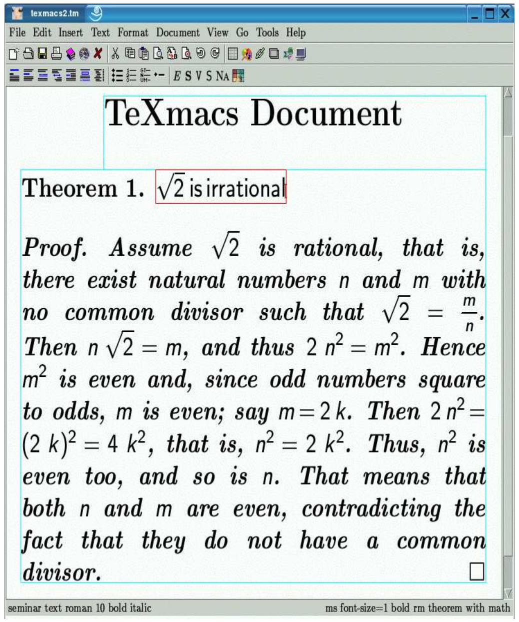
Different application scenarios call fro different specialisations and configurations

- Scenario A: Verified mathematical documents
- Scenario B: E-tutoring of proof in mathematics and engineering
- Scenario C: Formal hardware and software verification
- . . .

Scenario A: Verified Maths Documents

SFB 378 Project OMEGA:





texmacs2.tm

File Edit Insert Text Format Document View Go Tools Help

TeXmacs Document

Theorem 1. $\sqrt{2}$ is irrational

Proof. *Assume* $\sqrt{2}$ *is rational, that is, there exist natural numbers* n *and* m *with no common divisor such that* $\sqrt{2} = \frac{m}{n}$. *Then* $n\sqrt{2} = m$, *and thus* $2n^2 = m^2$. *Hence* m^2 *is even and, since odd numbers square to odds,* m *is even; say* $m = 2k$. *Then* $2n^2 = (2k)^2 = 4k^2$, *that is,* $n^2 = 2k^2$. *Thus,* n^2 *is even too, and so is* n. *That means that both* n *and* m *are even, contradicting the fact that they do not have a common divisor.* □

seminar text roman 10 bold italic

ms font-size=1 bold rm theorem with math

Mathematics

Assistance

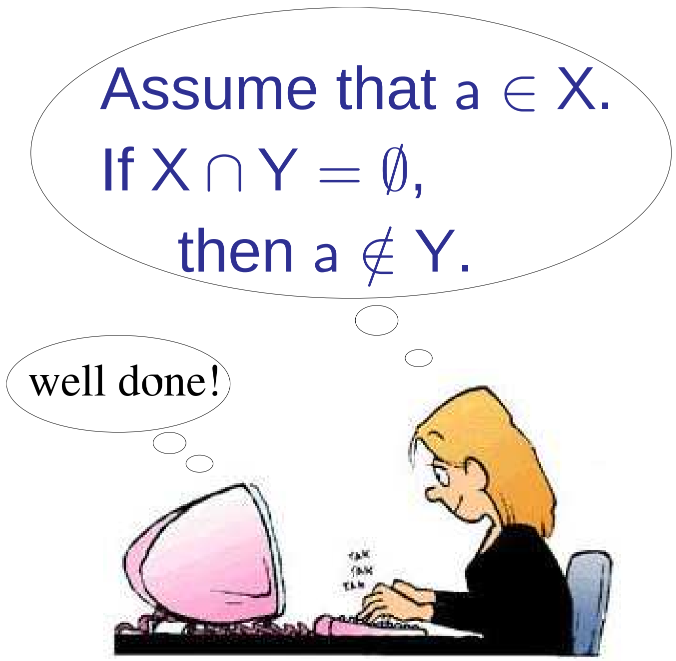
Environment

Scenario B: E-Tutoring

SFB 378 Project DIALOG:

Can we automate NL-based tutoring of mathematical proofs?

- Natural language analysis
- . . .
- Theorem Proving with a Mathematical Assistance System
- . . .
- Natural language generation

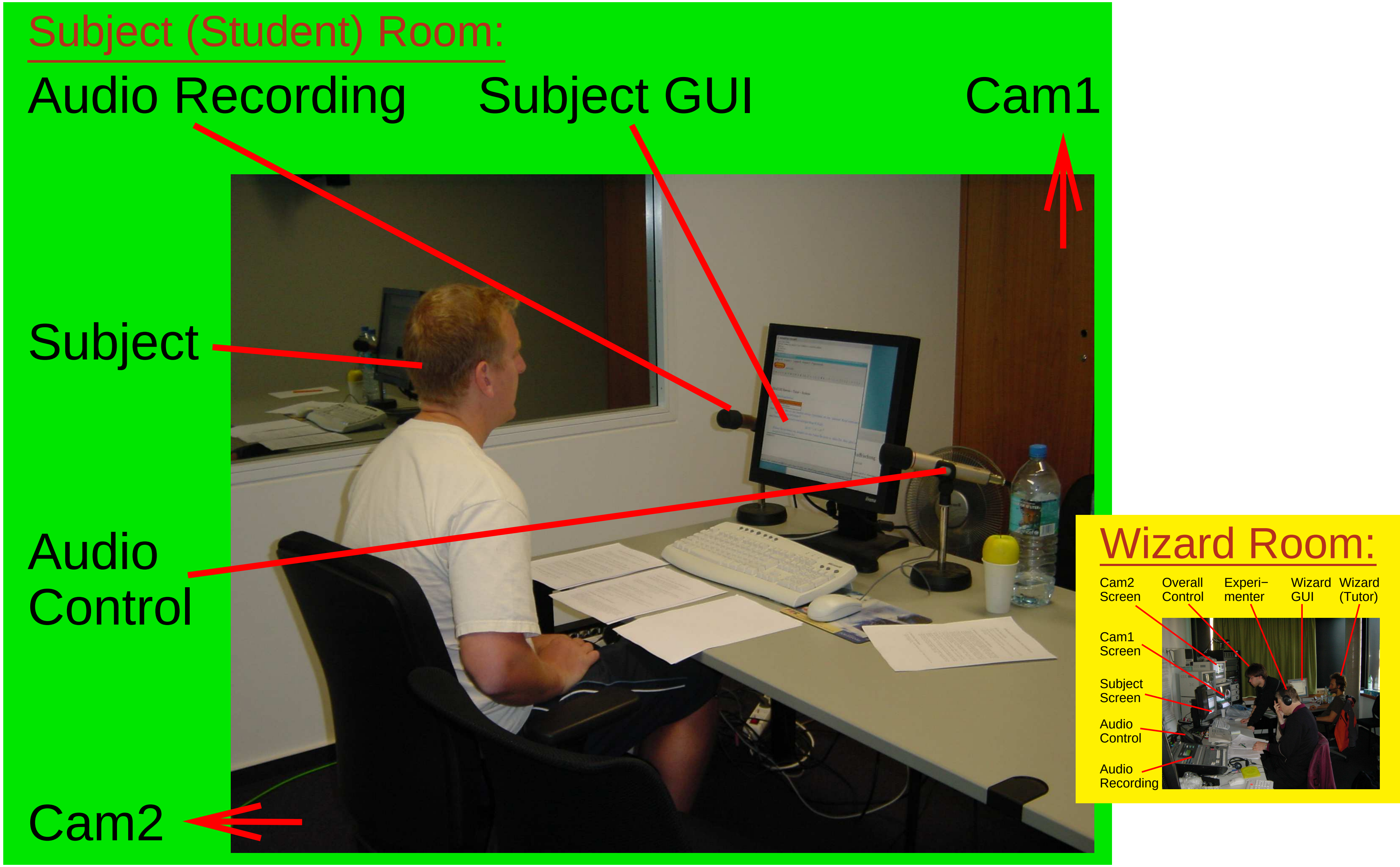


Big challenge:

Where to get real dialog examples from?

$\longrightarrow$ HCI experiments

Scenario B: Challenge for HCI



Scenario B: Challenge for HCI

Wizard (Tutor) Room:

Cam2 Screen

Overall Control

Experi– menter

Wizard GUI

Wizard (Tutor)

Cam1 Screen

Subject Screen

Audio Control

Audio Recording

Subject Room:

Audio Recording

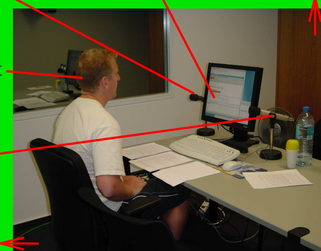
Subject GUI

Cam1

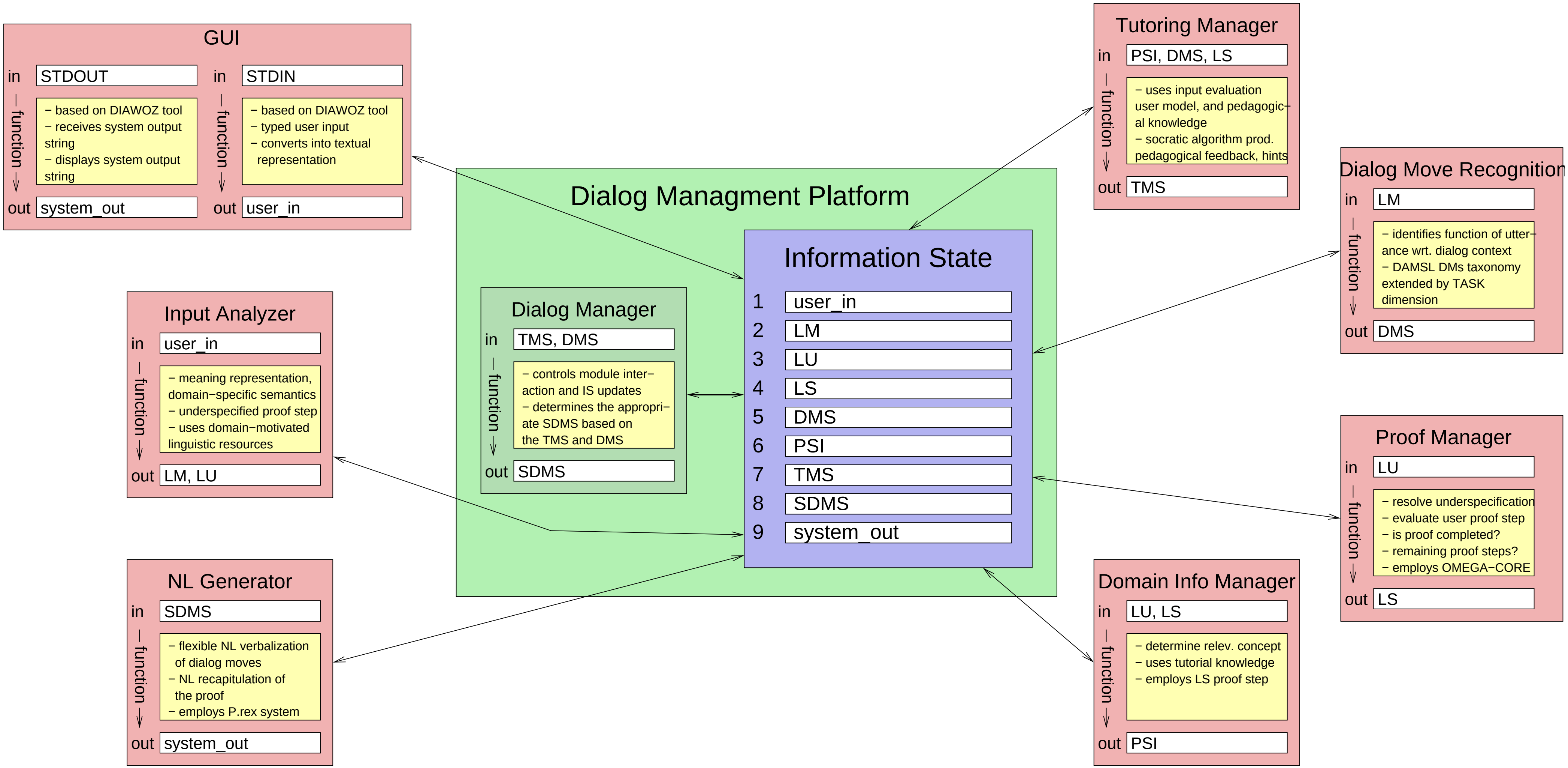
Subject

Audio Control

Cam2



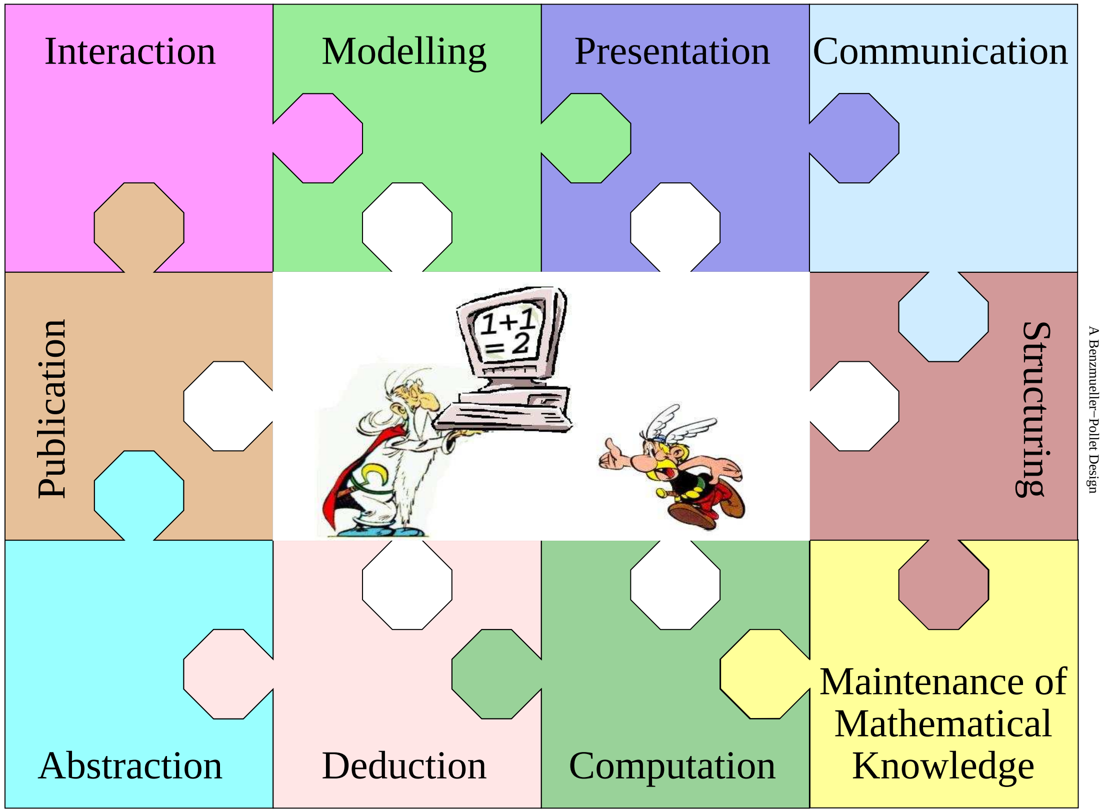
Scenario B: Need Further Components!



Why Service Oriented Architectures for Mathematics Assistence Systems?

Which System Architecture?

Integrated Mathematical Assistant Environment vs. ‘Pen-and-Paper’ Mathematics

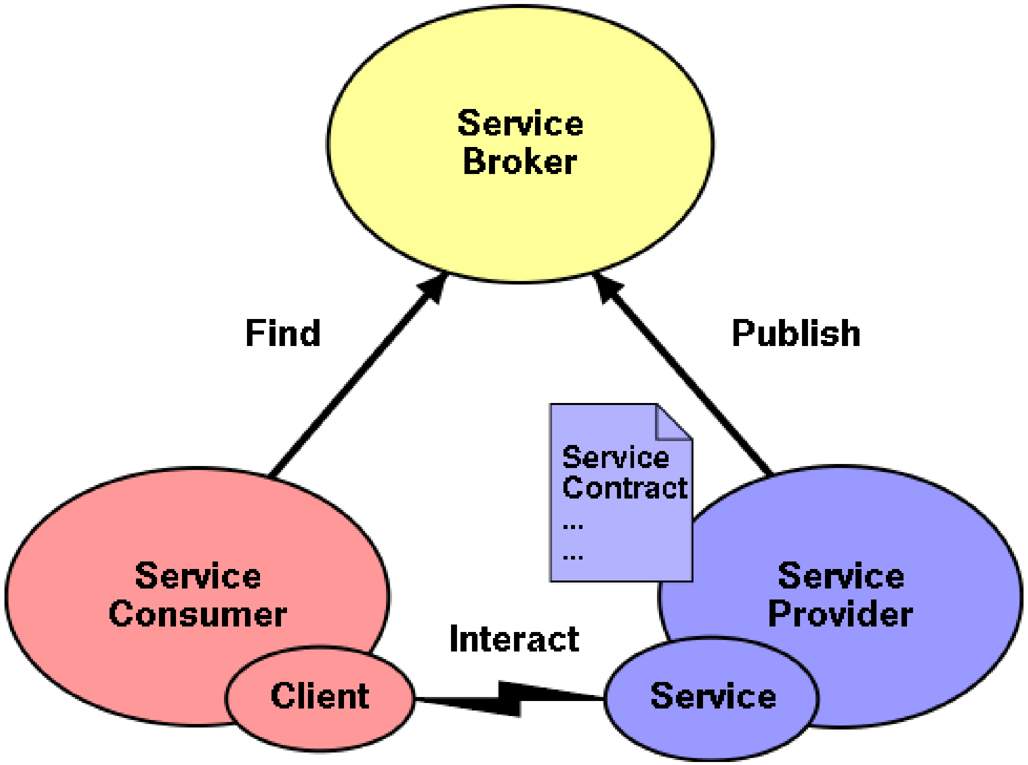


Which System Architecture?

Main Ideas

- all software components are modeled as services
- important
  - services have well-defined interfaces
  - services are discoverable
  - services may interoperate via loose coupling
- not important
  - the ’inner life’ of services
- focus of application design: composing services (using their well-defined interfaces) invoked over a network

Service Oriented Architectures (SOA)

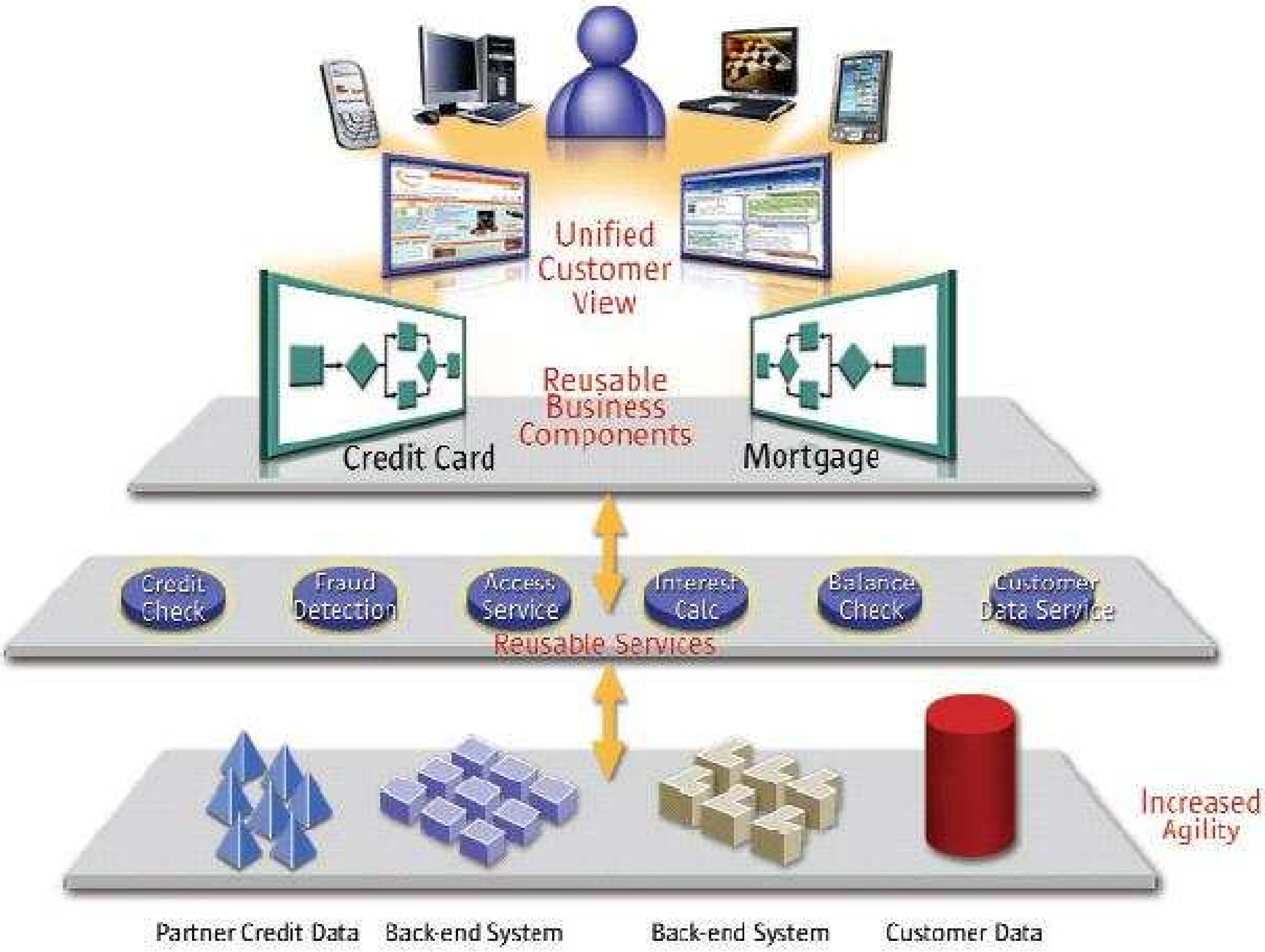


1. Service Provider makes service available (with Service Contract) and advertises it on Service Broker.

2. Service Consumer finds the compatible Service and its Service Contract via the Service Broker.

3. Service Consumer and Service Provider interact.

Service Oriented Architectures (SOA)



Benefits of SOA

Using SOA we want to achieve

- standalone and integrated components at the same time
- reusability
- logical and physical distribution
- interchangeability / replaceability
- high configurability and scalability
- availability and accessibility
- loose coupling with minimal interdependencies
- discoverability
- . . .

and make first steps towards a mathematical semantic web.

Own SOA Application/Technology:
From MathWeb to MathServe

Difficulties that motivated MathWeb

- Installation:
  - the (sub-)systems must be available for your OS
  - have to install the individual systems
  - may need to get a license
- Configuration:
  - must configure the MAS to your local installation
- Invocation:
  - need to know how the parameters to call each system
  - what are the “services” the systems can provide?
  - how to interpret the systems output (exit status / answer)

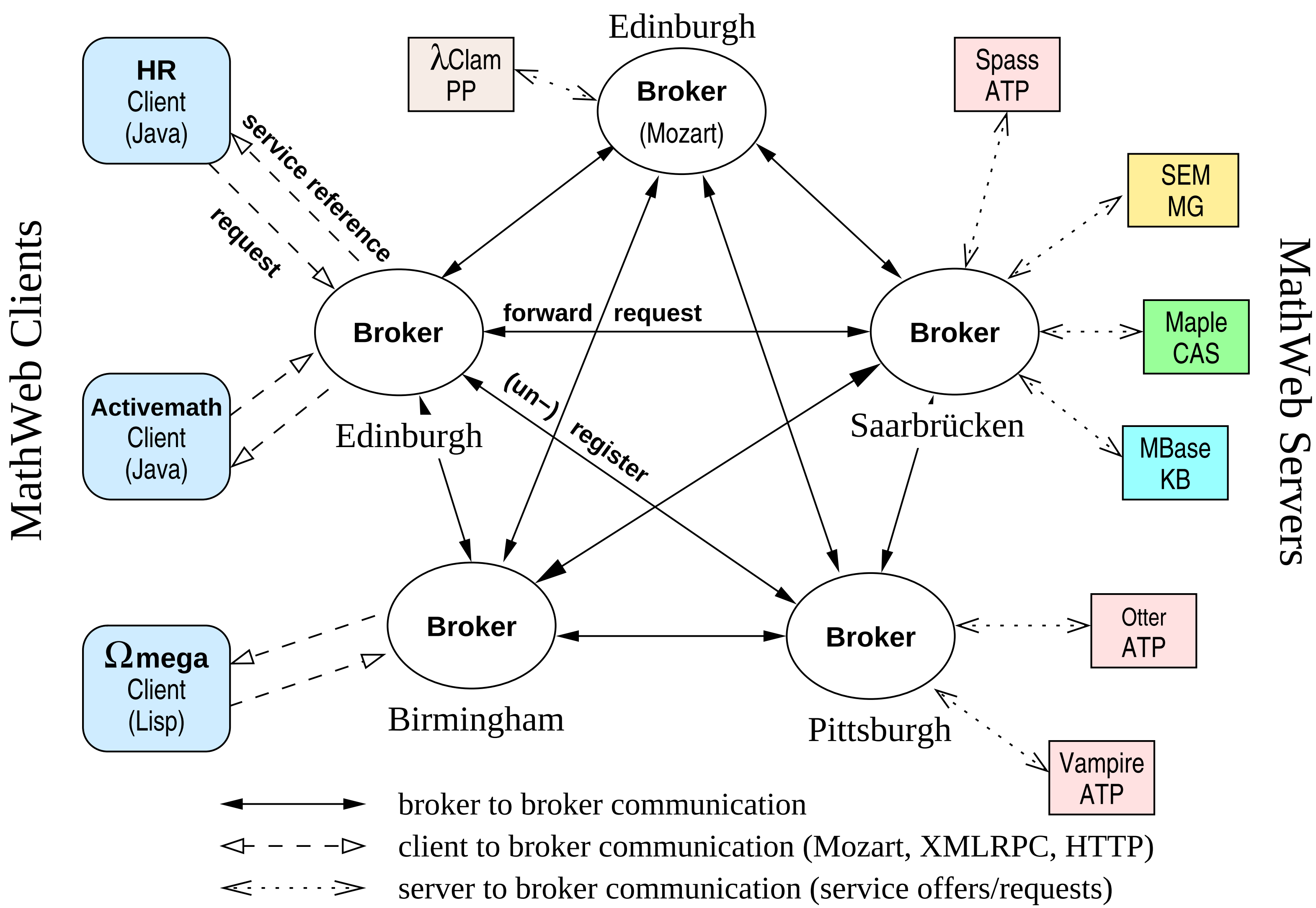
Difficulties that motivated MathWeb

- Language:
  - need to know the interface syntax for each system
  - need to write syntax transformations
  - are the transformations correct?

$\Rightarrow$ SOA provides a means to overcome these difficulties

(Developed our own SOA solution in 1996 without knowing about 'SOA')

The MathWeb



MathWeb Clients
MathWeb Servers
HR Client (Java)
Activemath Client (Java)
Ωmega Client (Lisp)
service reference
request
λClam PP
Edinburgh
Broker (Mozart)
Spass ATP
SEM MG
Maple CAS
MBase KB
Otter ATP
Vampire ATP
Broker
Edinburgh
forward request
(un-) register
Broker
Saarbrücken
Broker
Birmingham
Broker
Pittsburgh
broker to broker communication
client to broker communication (Mozart, XMLRPC, HTTP)
server to broker communication (service offers/requests)

Sample XML-RPC to MathWeb

```
<methodCall><methodName> Broker.getService</methodName>
 <params><param><value><string> SPASS</string></value></param></params>
</methodCall>

<methodCall><methodName> prove</methodName>
 <params><param><struct>
  <member><name>1</name><value><string>
   include('Axioms/EQU001+0.ax').
   include('Axioms/GRP004+0.ax').
   input_formula(conjecture118,conjecture,(! [B,C,D] :
    ((equal(inverse(B),C) & equal(multiply(C,B),D) ) <=>
     (equal(multiply(B,C),D) & equal(multiply(C,B),D) & equal(inverse(C),B))))
   </string></value></member>
   <member><name> syntax</name><value><string> tptp</string>
   <member><name> timeout</name><value><int> 40</int></value></member>
  </struct></param></params>
</methodCall>
```

From MathWeb to MathServe

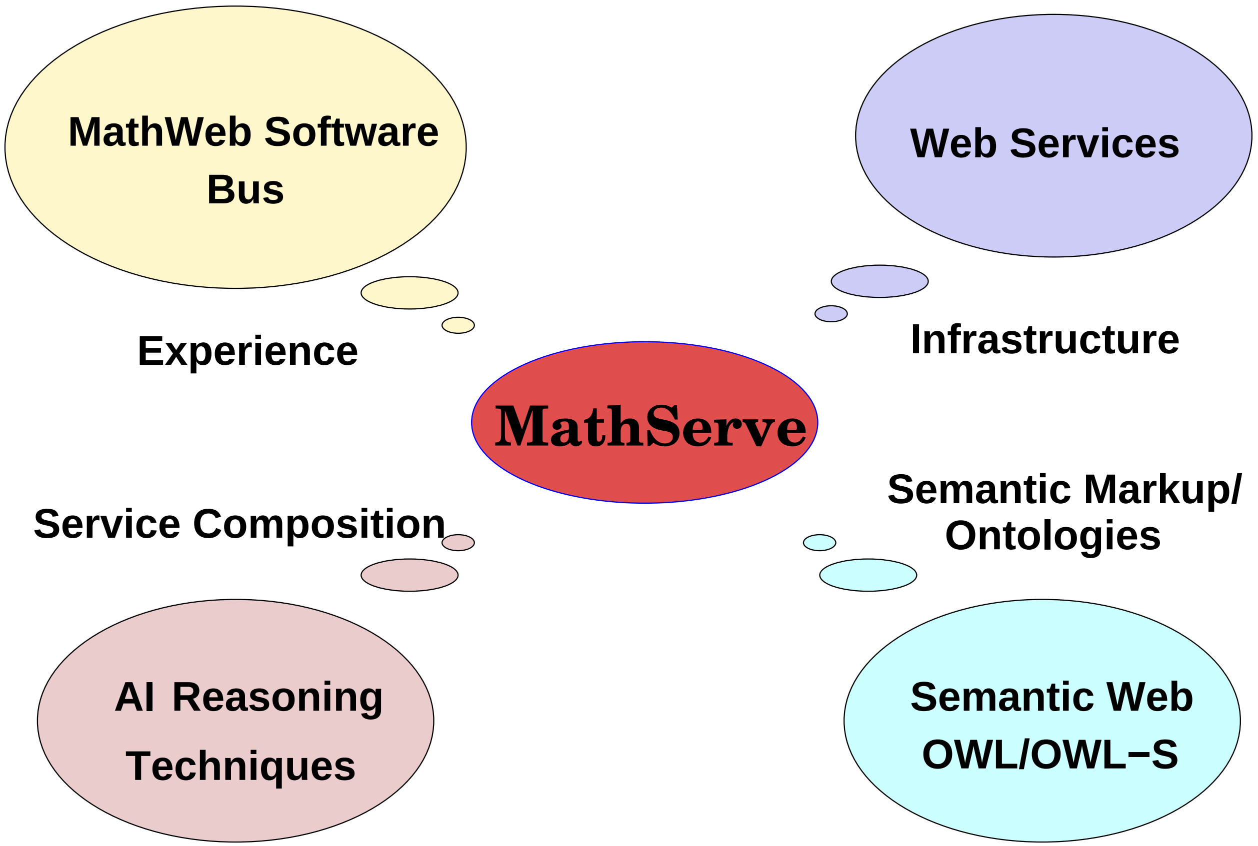
Away from the

system-oriented MathWeb

towards a

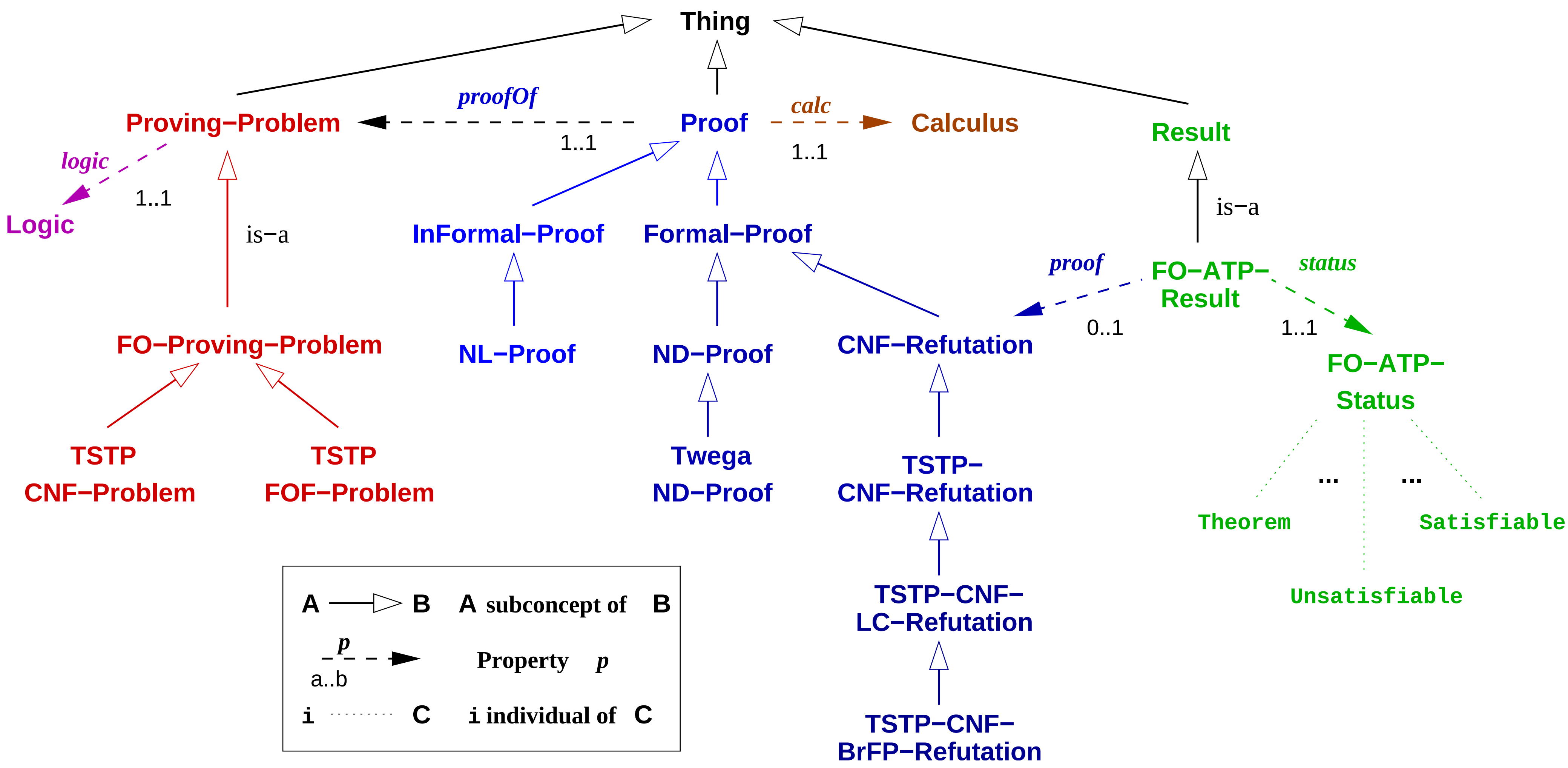
problem-oriented MathServe
(supporting also service compositions)

The MathServe System



Jürgen Zimmer (PhD)

OWL Ontology for Service Descriptions



OWL-S Atomic Process for ATP Service

The central part of a service description (example)

Service: EpATP	
input parameters:	*?problem*::TSTP-CNF-Problem
output parameters:	*?result*::FO-ATP-Result
pre-conditions:	$\top$
post-conditions:	*resultFor(?result, ?problem)*

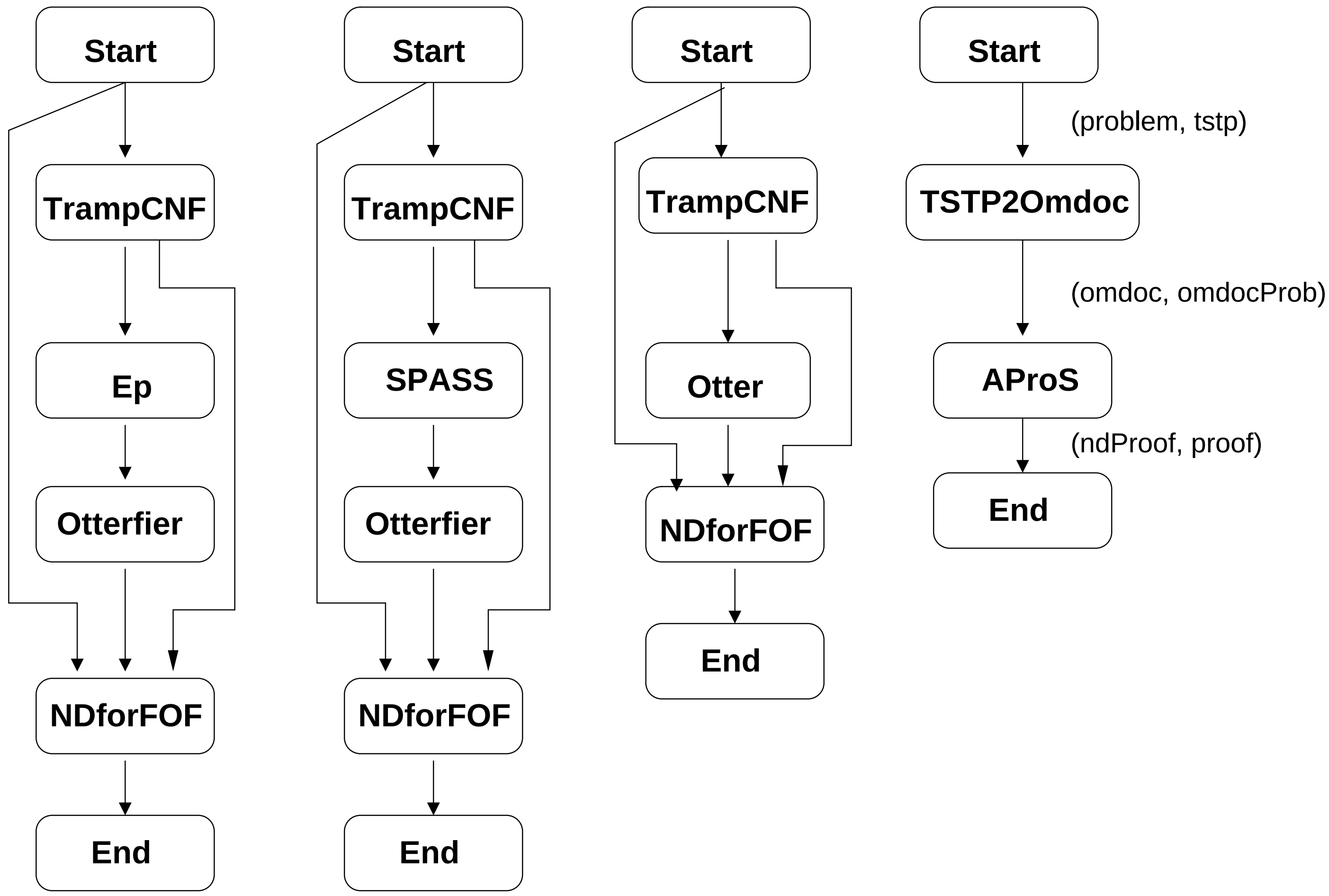
- we here completely omit XML details.
- conditions in Semantic Web Rule Language (SWRL)

The Service of Tramp

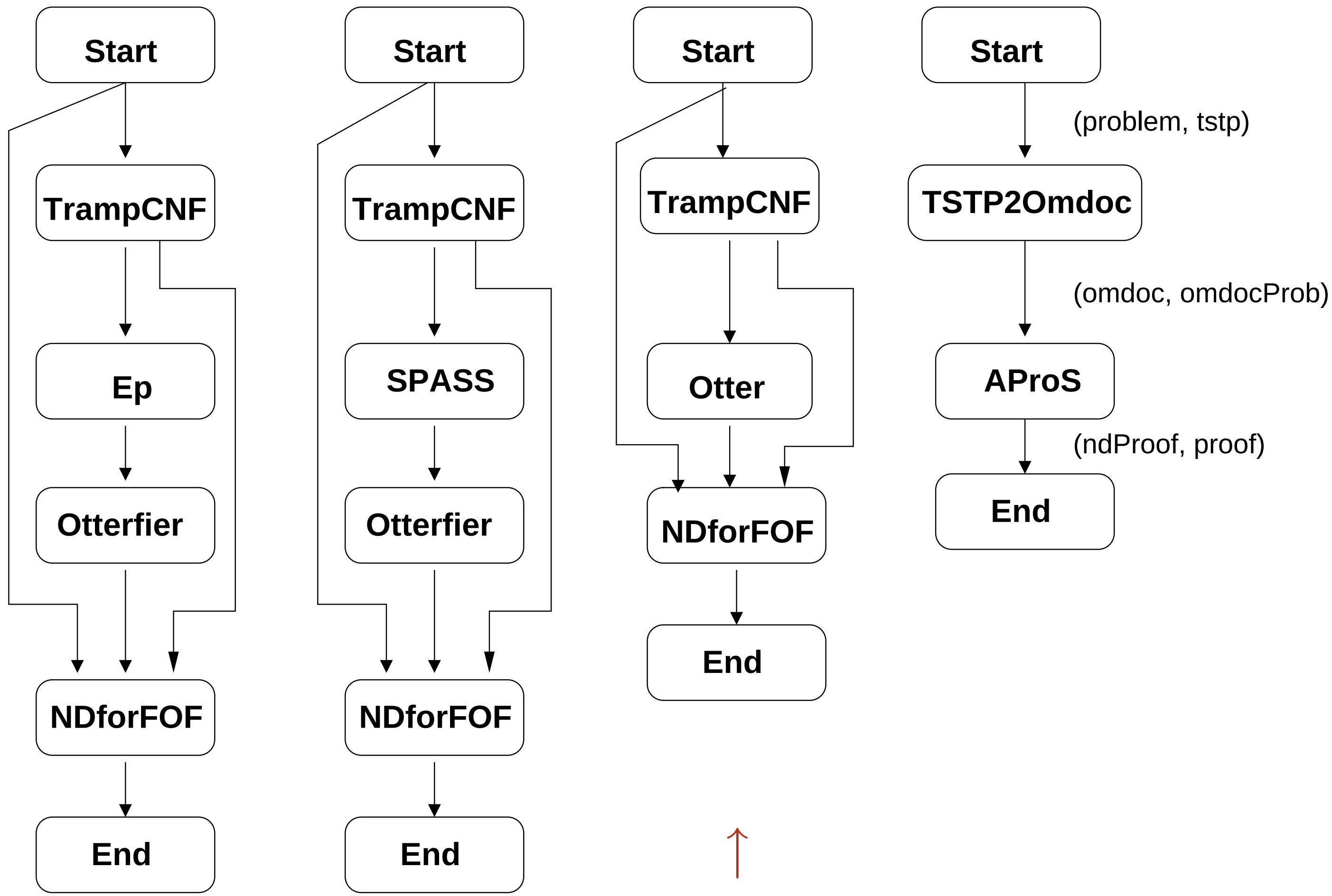
Service: Tramp-NDforFOF	
input parameters:	*?fofProblem*::TSTP-FOF-Problem *?atpResult*::FO-ATP-Result
output parameters:	*?ndProof*::Twega-ND-Proof
pre-conditions:	*resultFor(?atpResult, ?cnfProblem)* $\wedge$ *deltaRelated(?cnfProblem, ?fofProblem)*
post-conditions:	*proofOf(?ndProof, ?fofProblem)*

"*deltaRelated*": $\exists$ *mapping* δ: *Literals* $\mapsto$ *FOF-Atoms*

Planning finds many possible plans



Planning finds many possible plans



↑

Best Choice!

(based on probalistic reasoning)

How to become a World Champion with MathServe?

Evaluation on CASC-20

System	Problems given	Problems solved	Percentage of given	Percentage complete
MathServe 0.62	660	392	59.4%	59.4%
Ep 0.9pre3	660	409	62.0%	62.0%
Vampire 8.0	540	430	79.6%	65.2%
Vampire 7.0	300	262	87.4%	39.7%
MathServe 0.71	660	440	67.7%	67.7%

- MathServe 0.62 used older versions of EP (0.81) and Vampire (7.0).
- MathServe 0.62 could not handle large problems (> 2MB).
- MathServe 0.71 after CASC-20 with EP 0.9 and Vampire 8.0.

Availability and Usability

- Binary and source distribution available at

  http://www.ags.uni-sb.de/~jzimmer/mathserve.html

- MathServe is used by
  - Omega/Core proof assistant (University of Saarbrücken)
  - Hets toolset (University of Bremen)
  - Verifun project (Technical University Darmstadt)

See how MathServe 0.8 performs at the next CASC Championships

SOA and MathServ: Lecture Topics

- More background knowledge

SOA and MathServ: Lecture Topics

- More background knowledge
  (for better understanding of the MathServ application scenario)
  - maths
  - logic
  - ...

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies (as used in MathServ)
  - XML (communication infrastructure)
  - RDF (description of meta-data about web resources)
  - Description logics and OWL (express ontological structures)
  - WSDL (web service description language, supports no composition)
  - SOAP (message exchange protocol based on HTTP)
  - UDDI (standard for web service registries)
  - BPEL4WS (business processes execution language)
  - OWL-S (web service markup and description language, supports composition and grounding in WSDL)

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies
- Communication of Maths Content in the Semantic Web

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies
- Communication of Maths Content in the Semantic Web
  - MathML (standard for describing mathematical notation – structure + content)
  - OpenMath (standard for the representation and communication of mathematicl objects
  - OMDoc (semantic markup language for document structure and mathematical content)

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies
- Communication of Maths Content in the Semantic Web
- Composition of Web Services

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies
- Communication of Maths Content in the Semantic Web
- Composition of Web Services
  - AI planning techniques for service composition
  - Golog and the situation calculus
  - Markov decision procedures
  - Program synthesis

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies
- Communication of Maths Content in the Semantic Web
- Composition of Web Services
- Idea: Become a world champion with a group of students

SOA and MathServ: Lecture Topics

- More background knowledge
- State-of-the-art web service languages and technologies
- Communication of Maths Content in the Semantic Web
- Composition of Web Services
- Idea: Become a world champion with a group of students
  - Improve implementation of MathWeb with team of students
  - Participate at the yearly CASC competitions

Agent Oriented Architectures

This is a story for another lecture . . .