# ΩMEGA

# From Proof Planning towards Mathematical Knowledge Management

Serge Autexier[1,2] and Christoph Benzmüller[1]

[1] Saarland University, Saarbrücken, Germany

[2] German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany

September 25th, 2003

MKM Symposium, Heriot-Watt University, Edinburgh, Scottland

# Overview

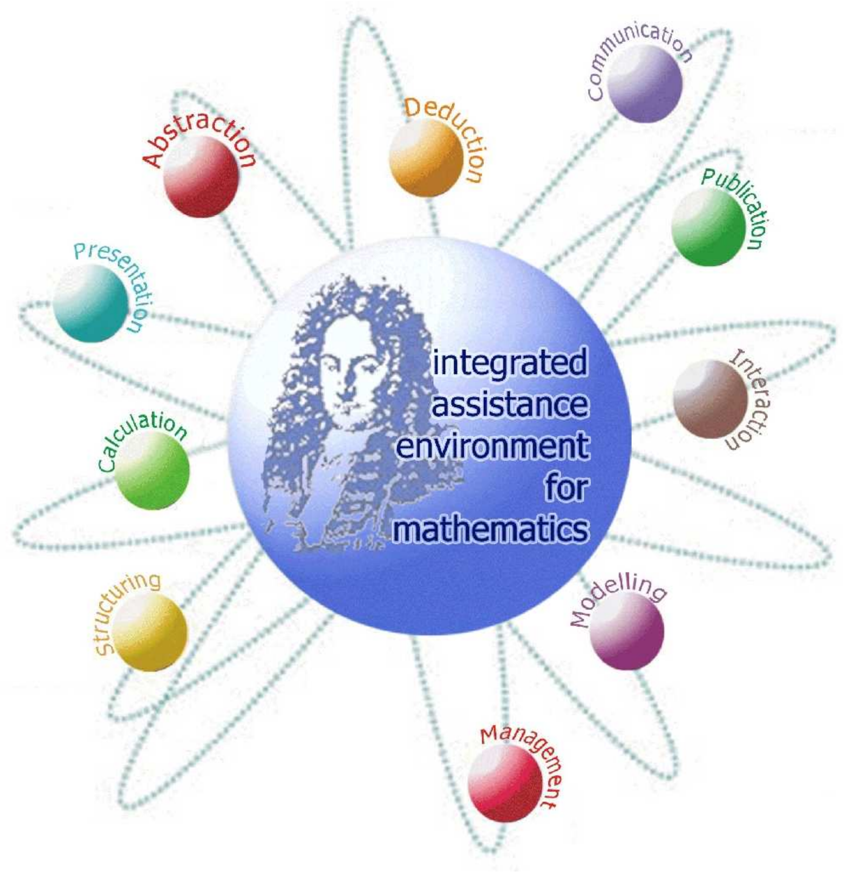
- Mathematical Assistant Environments
- The ΩMEGA Project
  - Mathematical Assistant In-the-small

    – research directions since early 90s –
  - Mathematical Assistant In-the-large

    – novel research directions –
  - Lessons Learned

Source: Autexier, Benzmüller

# Mathematical Assistant



CALCULEMUS-II illustration of MAs

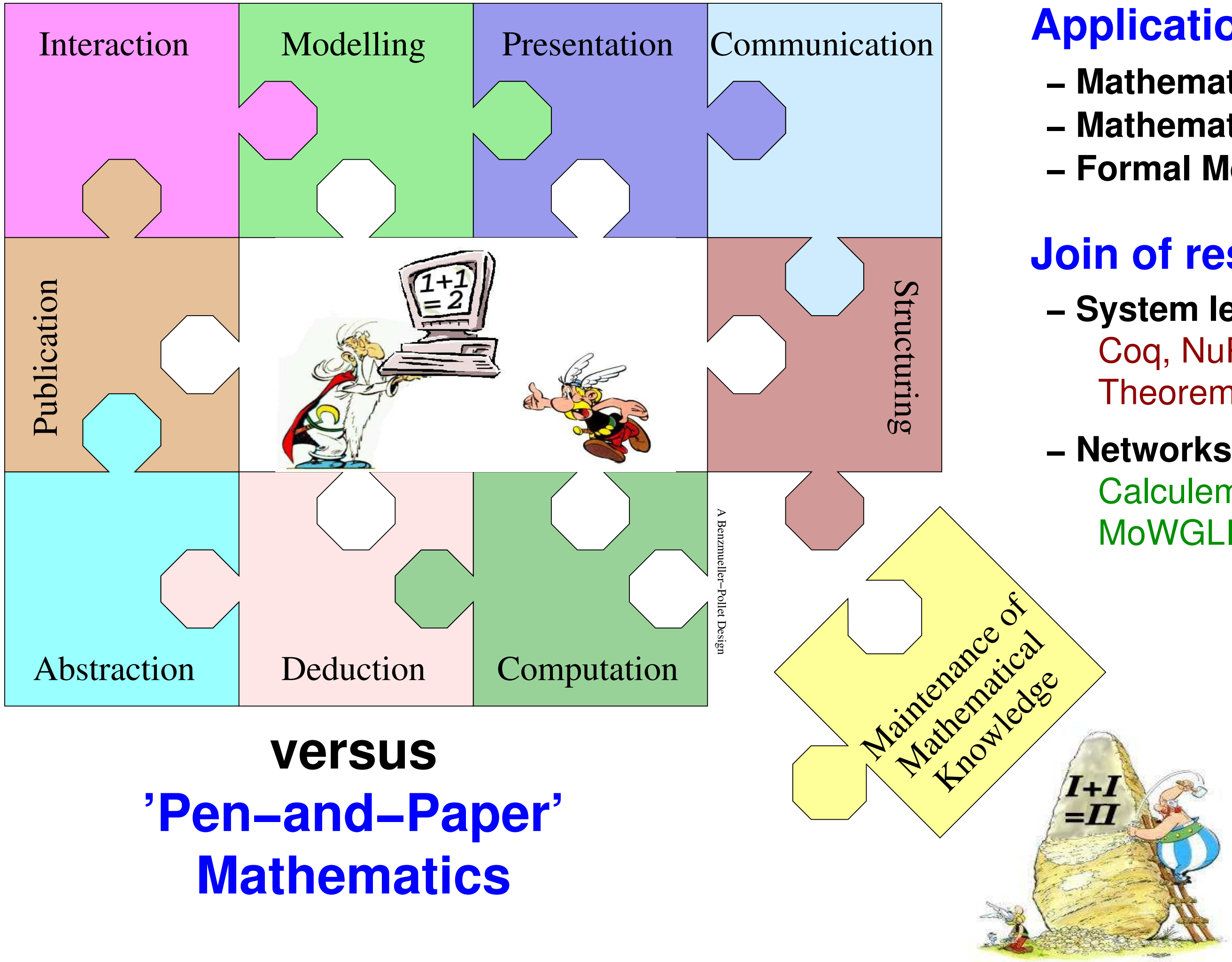
Mathematical Assistant (MA)

- Integrated computer-based support for most work tasks of a mathematician
- After enthusiasm of the 50s and 60s deduction systems area fragmented into subfields (similar to AI)
- Driving forces in reverting this trend:
  - MKM: top-down
  - CALCULEMUS: bottom-up

Source: Autexier, Benzmüller

# Mathematical Assistant

## Integrated
## Mathematical Assistance Environment



**versus**

## ’Pen–and–Paper’ Mathematics

## Applications

- Mathematics research
- Mathematics education
- Formal Methods, Bio–Informatics

## Join of ressources necessary

- **System level**
  Coq, NuPrl, Isabelle/HOL, PVS, Theorema, OMEGA, Clam, ...
- **Networks**
  Calculemus, MKM, Monet, MoWGLI, ...

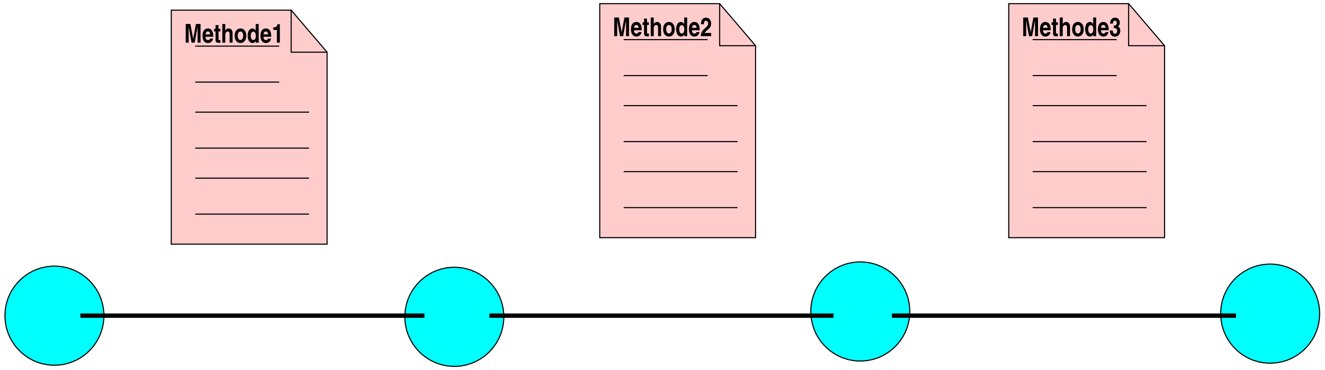
Source: Autexier, Benzmüller

# Mathematical Assistant In-the-small

# Research directions in the ΩMEGA project since the early 90s

Source: Autexier, Benzmüller

# Proof Planning



ΩMEGA born in early 90s; inspired by [Bundy88]

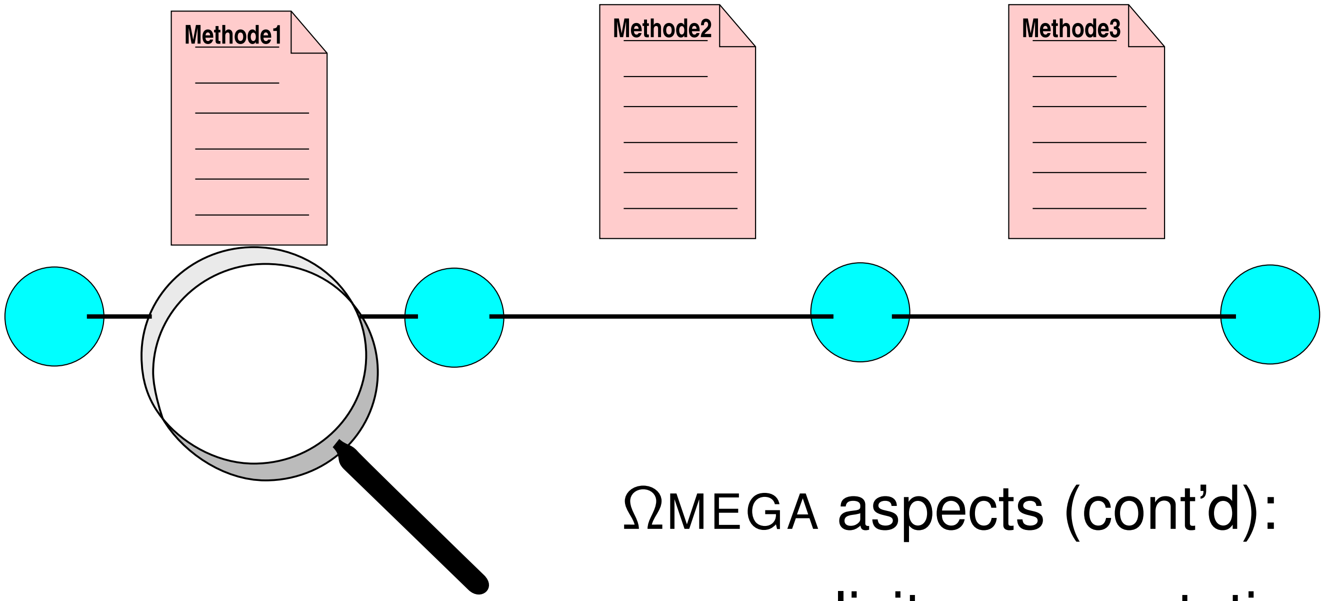
- paradigm shift from classical FOL ATP to proof planning in HOL

ΩMEGA aspects:

- declarative, domain specific control layer
- strategy = domain specific instantiation of a general proof search algorithm with set of proof methods and control information
- multi-strategy proof planning

Source: Autexier, Benzmüller

# Proof Planning

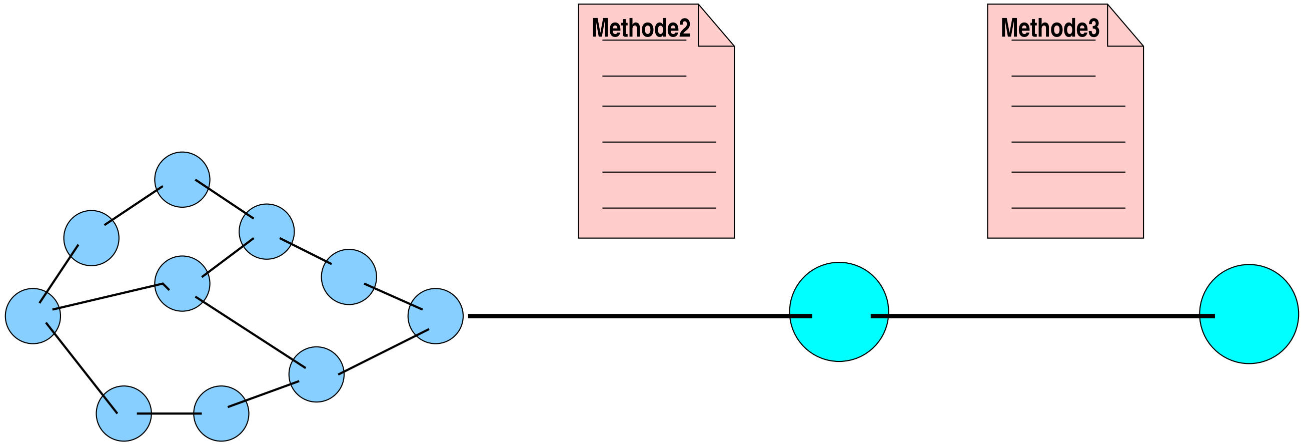


ΩMEGA aspects (cont'd):

- explicit representation of proof plans
- under-specification of pre-conditions: potentially non-sound proof plans
- soundness guaranteed via . . .

Source: Autexier, Benzmüller

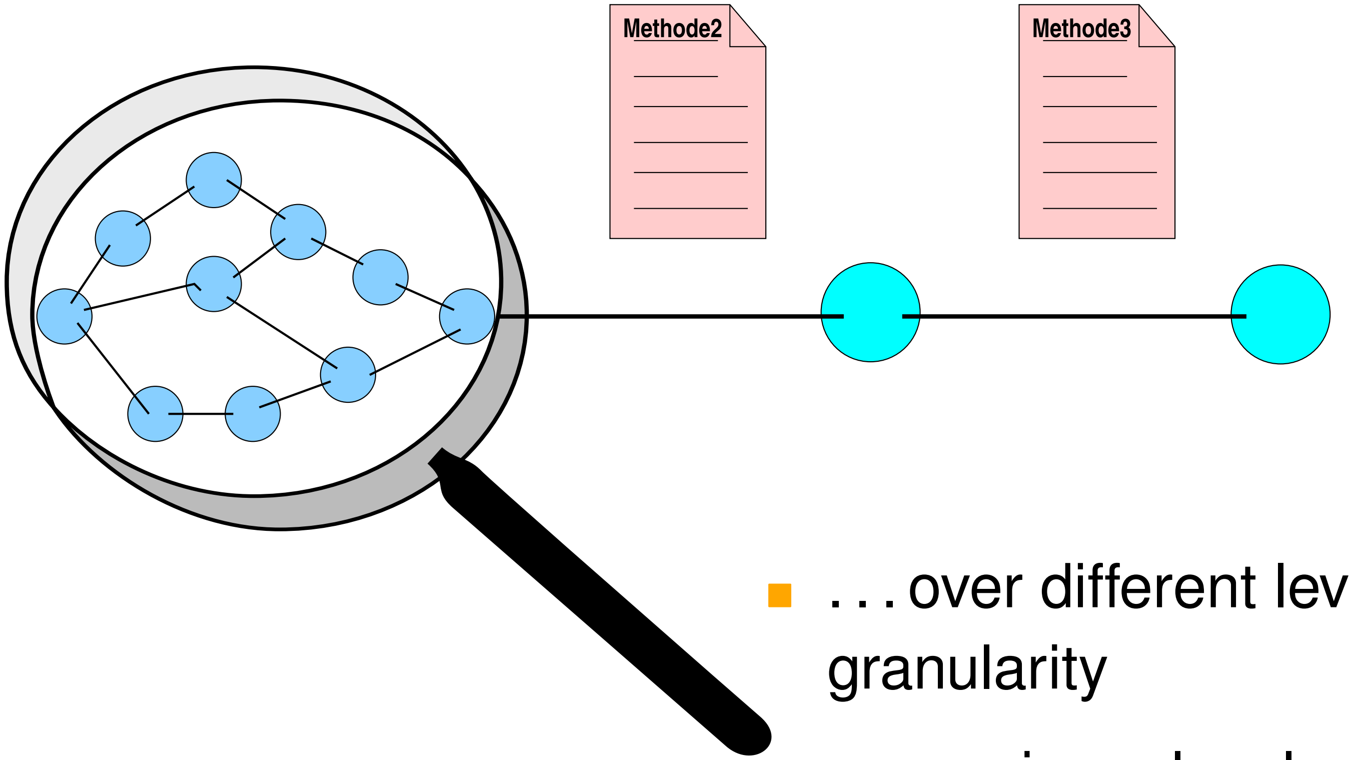
# Proof Planning



- . . . proof (plan) expansion over . . .

Source: Autexier, Benzmüller

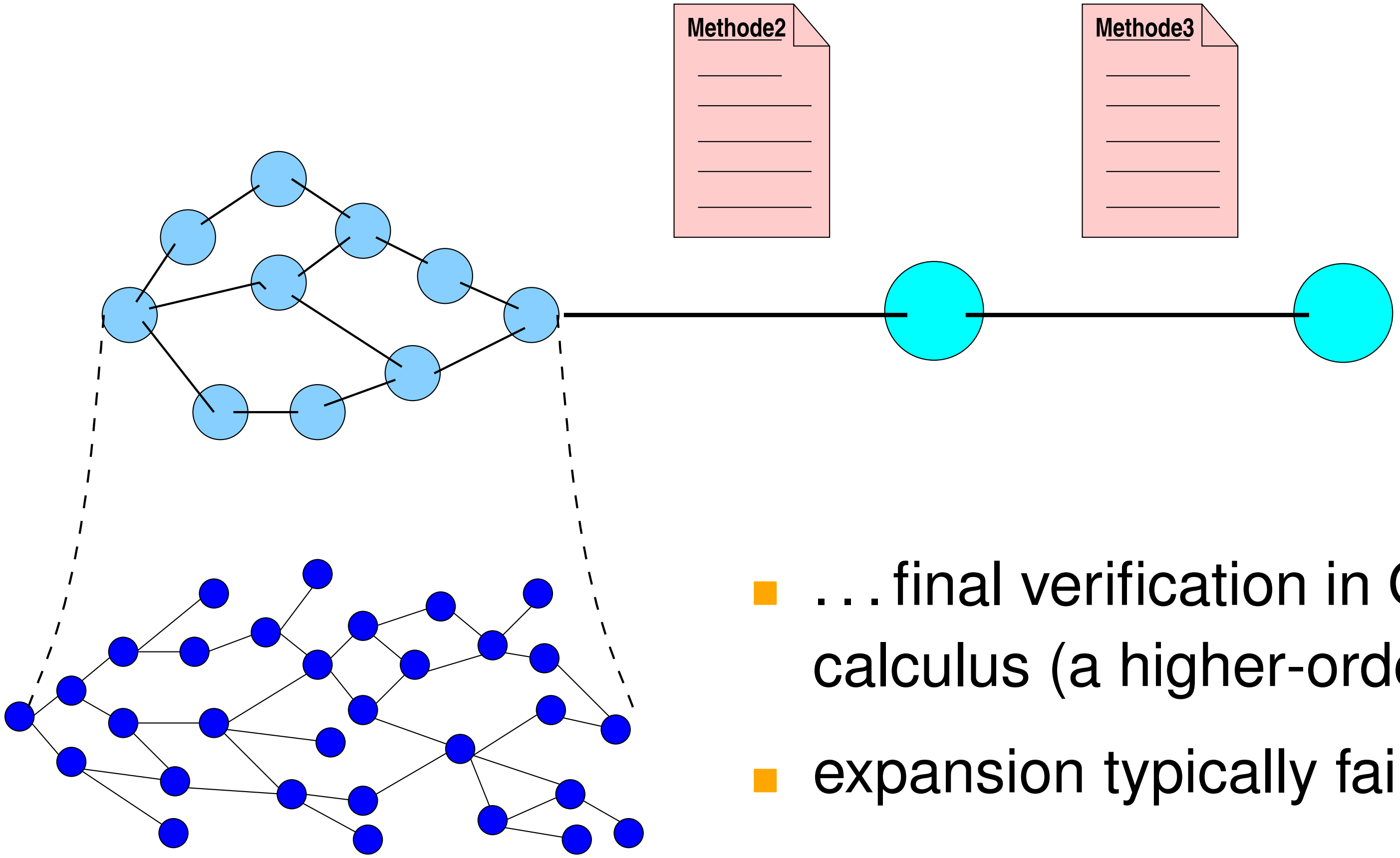
# Proof Planning



- . . . over different levels of granularity
- expansion = local proof search / proof construction problem
- may employ external reasoners

Source: Autexier, Benzmüller

# Proof Planning



- . . . final verification in OMEGAs base calculus (a higher-order ND variant)
- expansion typically fails early!

Source: Autexier, Benzmüller

# Proof Planning

Main References

[MelisSiekmann-AIJ-99] Knowledge-based proof planning

[Meier-Diss-03] Multi-strategy proof planning

[MeierETAL-JSC-02, CohenEtAl-CADE-03, SiekmannEtAl-35yAutomath-03]
Proof planning with external specialist reasoners

Discussion

+ problem classes in specific domains; coordination of systems

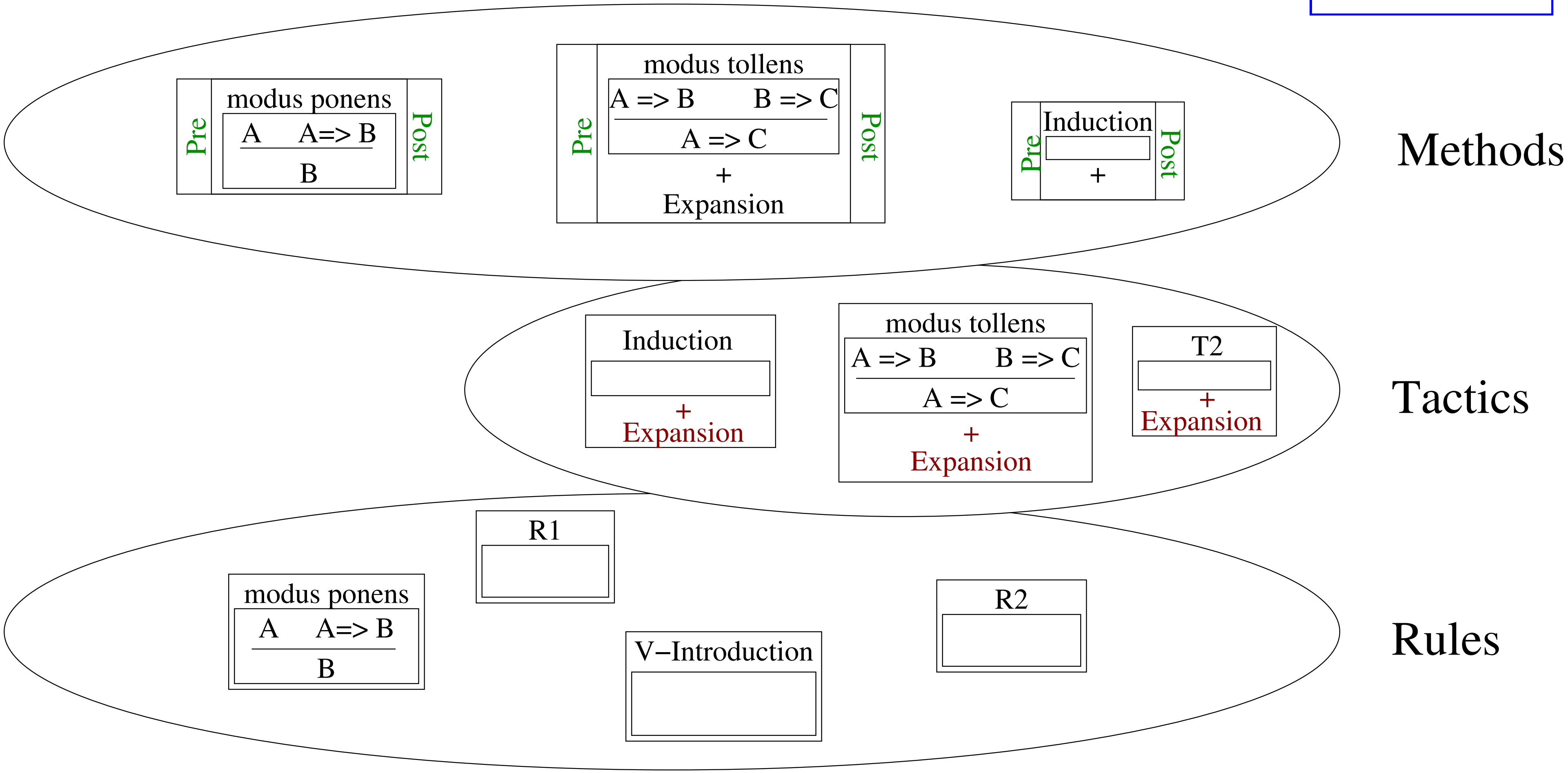
– Brittleness and logic layer dependency; mixed-iniative PP

New Directions

⇒ Proof planning based on CORE (see 2nd part of talk)

Source: Autexier, Benzmüller

# Interactive Proof



Differences to LCF: explicit representation (delayed expansion), potentially non-sound tactics and methods, verification via expansion

Source: Autexier, Benzmüller

# Interactive Proof

*Theorem:* $\sqrt{2}$ is irrational.

*Proof:* (by contradiction)
Assume $\sqrt{2}$ is rational, that is, there exist natural numbers $m, n$ with no common divisor such that $\sqrt{2} = m/n$. 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.

- declarative style of argumentation:
  from assertions A and B follows C
- logic layer (e.g. a la ND- or Sequent-Calculus) treated implicit

⇒ mismatch between procedural style logic-level reasoning as employed in todays theorem provers and declarative assertion level reasoning as typical for mathematical texts

Source: Autexier, Benzmüller

# Interactive Proof

Declarative approach versus procedural approach

Network of proof ‘islands’

$$\dfrac{\dfrac{2 * n^2 = m^2}{\mathsf{Even}(m^2)}\;\text{Island}}{\mathsf{Even}(m)}\;\text{Island}$$
$$\vdots$$

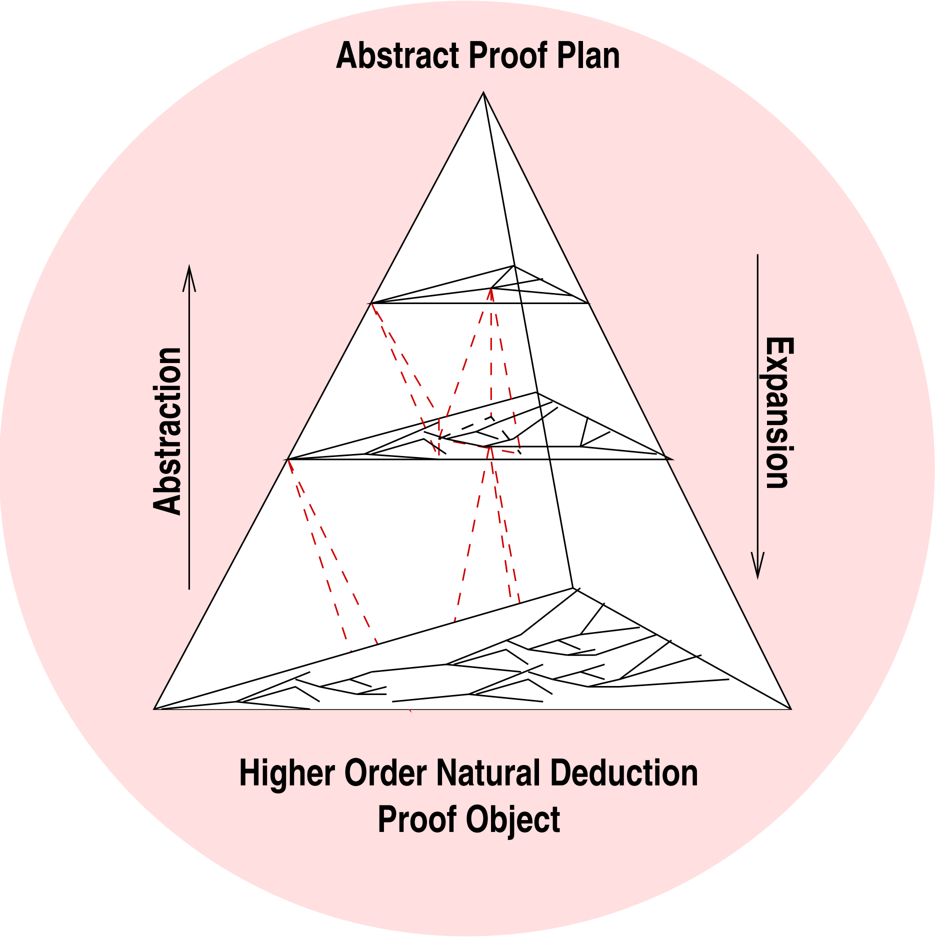
- Islands structure the proof in natural form
- Islands provide no argument for soundness

⇒ Verification: expansion of island steps (automated, interactive, recursive island approach)

Source: Autexier, Benzmüller

# Proof Data Structure

ΩMEGA PDS



- Maintenance of proof developments at
- different layers of granularity which are
- connected to each other

Source: Autexier, Benzmüller

# Proof Data Structure

Main References

[CheikhrouhouSorge-ACIDCA-00] Overview on PDS

[SiekmannEtAl-35yAutomath] Working with PDS

Discussion

+ Support for proof development at different levels of granularity, proof expansion and contraction, non-soundness and verification

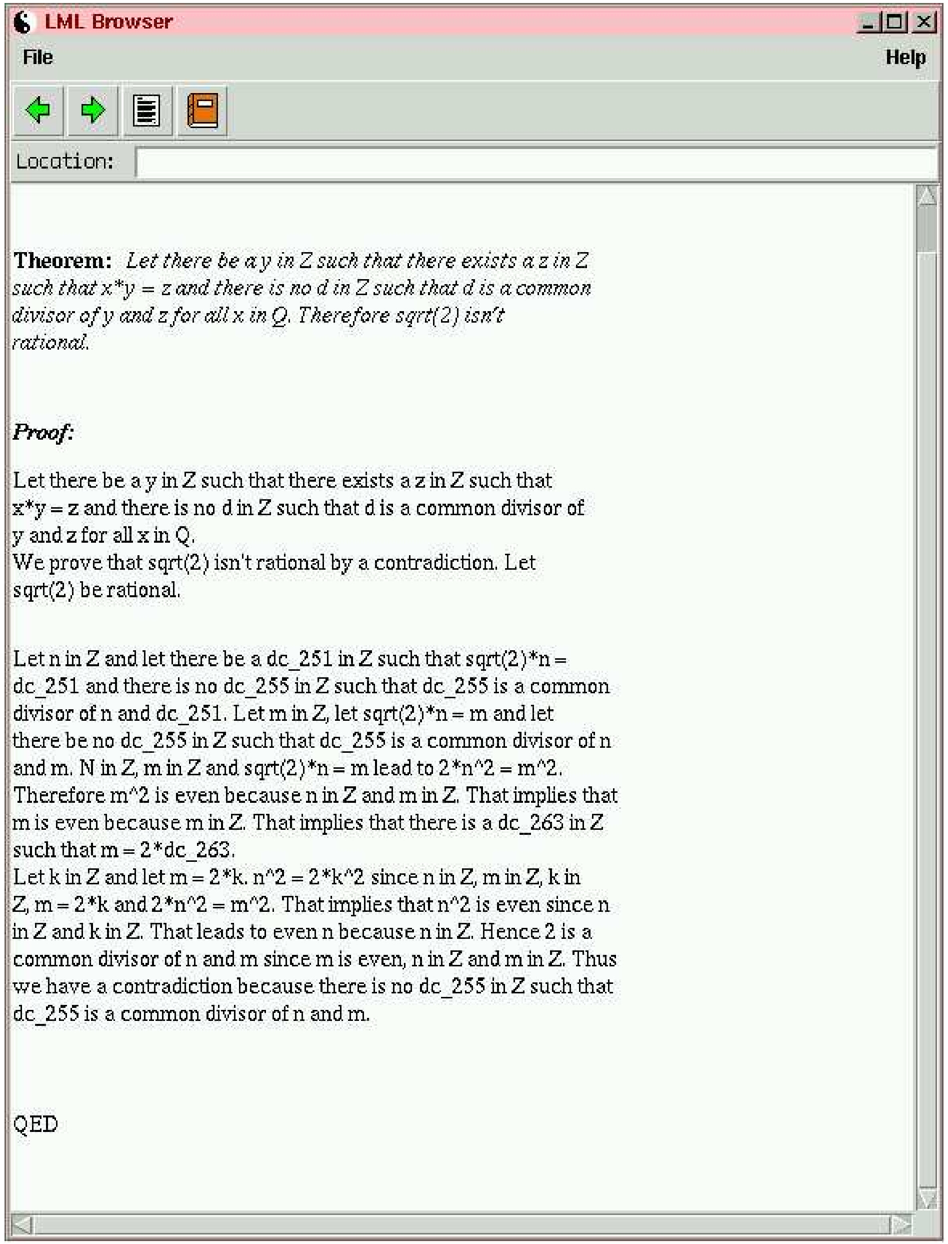
– Missing: support for change of representation language

New Directions

⇒ A PDS for different levels of granularity and representational abstraction [AutexierBenmüllerHutter-SEKI-03]

Source: Autexier, Benzmüller

# Proof Verbalization



**Theorem:** *Let there be a y in Z such that there exists a z in Z such that x*y = z and there is no d in Z such that d is a common divisor of y and z for all x in Q. Therefore sqrt(2) isn't rational.*

***Proof:***

Let there be a y in Z such that there exists a z in Z such that x*y = z and there is no d in Z such that d is a common divisor of y and z for all x in Q.
We prove that sqrt(2) isn't rational by a contradiction. Let sqrt(2) be rational.

Let n in Z and let there be a dc_251 in Z such that sqrt(2)*n = dc_251 and there is no dc_255 in Z such that dc_255 is a common divisor of n and dc_251. Let m in Z, let sqrt(2)*n = m and let there be no dc_255 in Z such that dc_255 is a common divisor of n and m. N in Z, m in Z and sqrt(2)*n = m lead to 2*n^2 = m^2. Therefore m^2 is even because n in Z and m in Z. That implies that m is even because m in Z. That implies that there is a dc_263 in Z such that m = 2*dc_263.
Let k in Z and let m = 2*k. n^2 = 2*k^2 since n in Z, m in Z, k in Z, m = 2*k and 2*n^2 = m^2. That implies that n^2 is even since n in Z and k in Z. That leads to even n because n in Z. Hence 2 is a common divisor of n and m since m is even, n in Z and m in Z. Thus we have a contradiction because there is no dc_255 in Z such that dc_255 is a common divisor of n and m.

QED

P.REX (successor of PROVERB):

- lifting of proofs in the PDS to assertion level
- macro-planning text structure
- micro-planning sentence structure and linguistic realization
- generation of natural language representation
- pre-required: linguistic knowledge
- user-adaptive proof explanation

Source: Autexier, Benzmüller

# Proof Verbalization

Main References

| | |
|---|---|
| [Huang-CADE-94] | PROVERB, Assertion Level |
| [Fiedler-IJCAR-01, Fiedler-PhD-01] | P.REX, proof explanation |

Discussion

+ Flexible, adaptable, non-template based proof verbalization

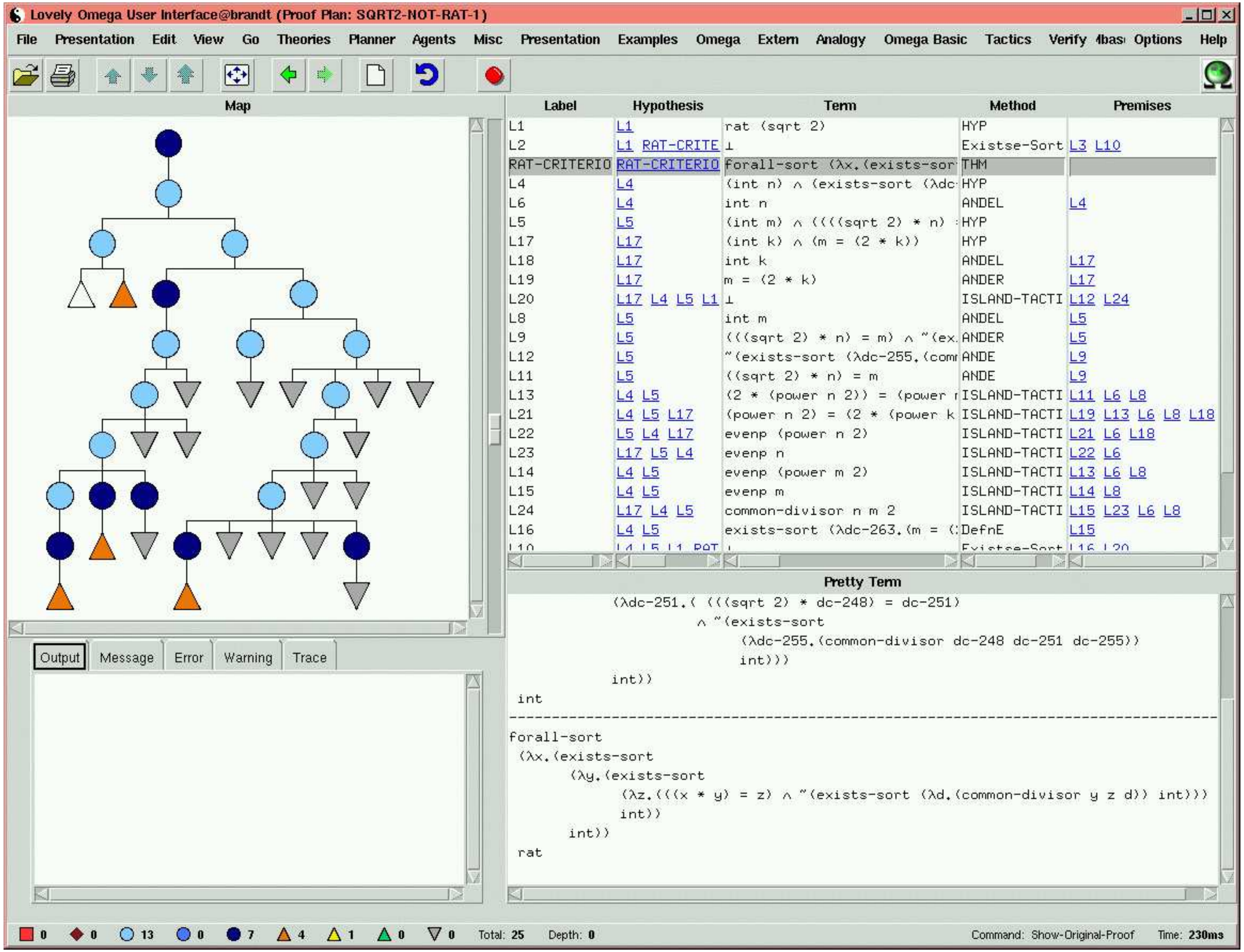
– Missing: Full natural language DIALOG at assertion level

New Directions

⇒ DIALOG project (see 2nd part of talk and talk on ’Assertion level proofs with under-specification’)

Source: Autexier, Benzmüller

# User Interface



| Label | Hypothesis | Term | Method | Premises |
|---|---|---|---|---|
| L1 | L1 | rat (sqrt 2) | HYP | |
| L2 | L1 RAT-CRITE | ⊥ | Existse-Sort | L3 L10 |
| RAT-CRITERIO | RAT-CRITERIO | forall-sort (λx.(exists-sor | THM | |
| L4 | L4 | (int n) ∧ (exists-sort (λdc | HYP | |
| L6 | L4 | int n | ANDEL | L4 |
| L5 | L5 | (int m) ∧ ((((sqrt 2) * n) : | HYP | |
| L17 | L17 | (int k) ∧ (m = (2 * k)) | HYP | |
| L18 | L17 | int k | ANDEL | L17 |
| L19 | L17 | m = (2 * k) | ANDER | L17 |
| L20 | L17 L4 L5 L1 | ⊥ | ISLAND-TACTI | L12 L24 |
| L8 | L5 | int m | ANDEL | L5 |
| L9 | L5 | (((sqrt 2) * n) = m) ∧ ~(ex. | ANDER | L5 |
| L12 | L5 | ~(exists-sort (λdc-255.(comr | ANDE | L9 |
| L11 | L5 | ((sqrt 2) * n) = m | ANDE | L9 |
| L13 | L4 L5 | (2 * (power n 2)) = (power r | ISLAND-TACTI | L11 L6 L8 |
| L21 | L4 L5 L17 | (power n 2) = (2 * (power k | ISLAND-TACTI | L19 L13 L6 L8 L18 |
| L22 | L5 L4 L17 | evenp (power n 2) | ISLAND-TACTI | L21 L6 L18 |
| L23 | L17 L5 L4 | evenp n | ISLAND-TACTI | L22 L6 |
| L14 | L4 L5 | evenp (power m 2) | ISLAND-TACTI | L13 L6 L8 |
| L15 | L4 L5 | evenp m | ISLAND-TACTI | L14 L8 |
| L24 | L17 L4 L5 | common-divisor n m 2 | ISLAND-TACTI | L15 L23 L6 L8 |
| L16 | L4 L5 | exists-sort (λdc-263.(m = (: | DefnE | L15 |
| L10 | L4 L5 L1 RAT | ⊥ | Existse-Sort | L16 L20 |

Pretty Term

```
                (λdc-251.( (((sqrt 2) * dc-248) = dc-251)
                               ∧ ~(exists-sort
                                      (λdc-255.(common-divisor dc-248 dc-251 dc-255))
                                      int)))
                int))
 int
------------------------------------------------------------------------------------------
forall-sort
 (λx.(exists-sort
        (λy.(exists-sort
               (λz.(((x * y) = z) ∧ ~(exists-sort (λd.(common-divisor y z d)) int)))
               int))
        int))
 rat
```

■ 0 ◆ 0 ○ 13 ○ 0 ● 7 △ 4 △ 1 △ 0 ▽ 0 Total: 25 Depth: 0 Command: Show-Original-Proof Time: 230ms

Source: Autexier, Benzmüller

# User Interface

Main References

[SiekmannEtAl-99] LOUI: Lovely OMEGA User Interface

Discussion

+ Support for different (connected) views on proof developments: linearized ND style, proof tree (PDS), natural language

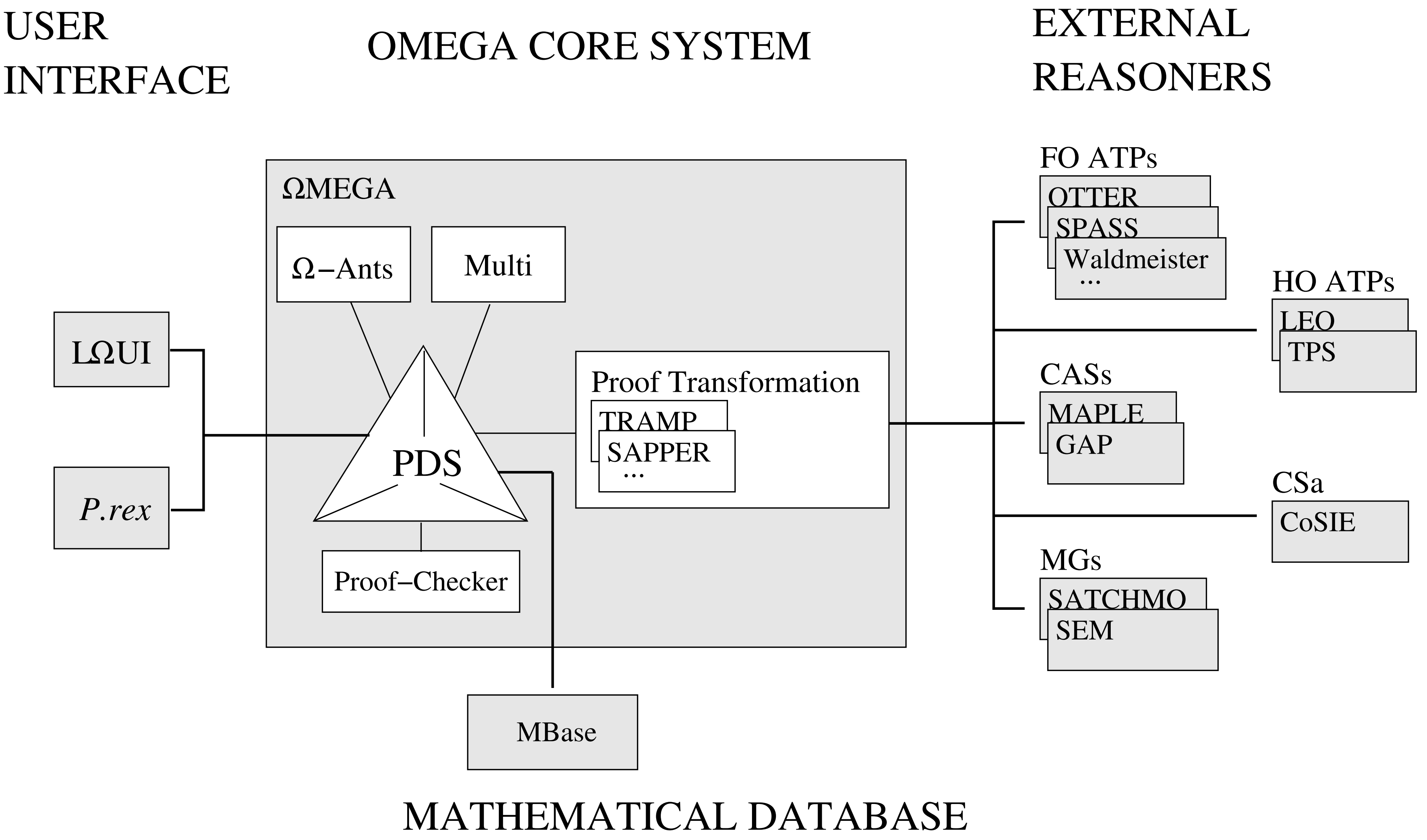
– What do users really want to see? Which users?

– Missing: optimal, integrated support for other mathematical activities such as publication, authoring, modeling, etc.

Source: Autexier, Benzmüller

# Mathematical Knowledge

ΩMEGA



Source: Autexier, Benzmüller

UNIVERSITÄT DES SAARLANDES

# Mathematical Knowledge

Main References

| | |
|---|---|
| [FrankeKohlhase-CADE-00] | MBASE mathematical knowledge base |
| [Kohlhase-AISC-00,Kohlhase-03] | OMDoc |

Discussion

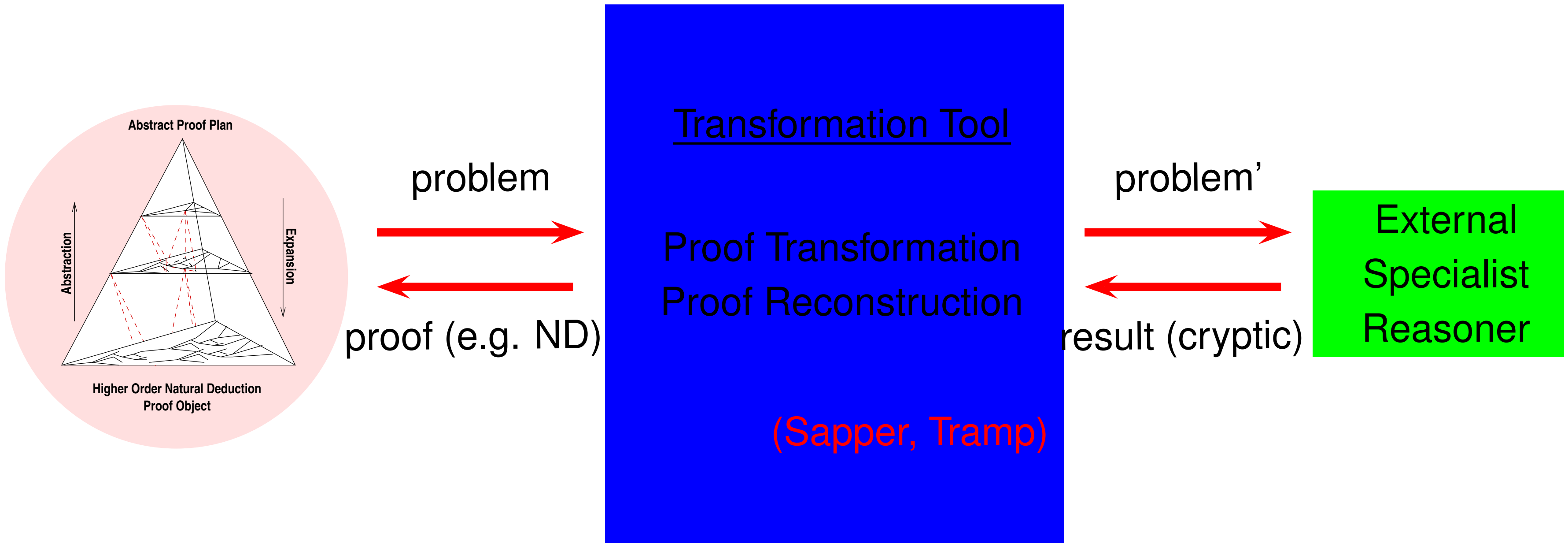
- + first step towards system independence
- – still dependable on logic context
- – version control: concurrent, joint development of mathematical knowledge
- – system independent representation formats for proof rules, tactics, methods, and control knowledge

Source: Autexier, Benzmüller

# External Specialist Reasoners

Usually required in OMEGA:

- white box integration of external specialist reasoners
- tools for extraction and transformation of results



problem

proof (e.g. ND)

Transformation Tool

Proof Transformation

Proof Reconstruction

(Sapper, Tramp)

problem'

result (cryptic)

External Specialist Reasoner

Source: Autexier, Benzmüller

# External Specialist Reasoners

Main References

| | |
|---|---|
| [Meier-CADE-00] | TRAMP: Integration of FOL ATPs into OMEGA |
| [Sorge-FROCOS-00] | SAPPER: Integration of CAS into OMEGA |
| [BenzmüllerEtAl-99] | Integration of TPS into OMEGA |
| [MelisEtAl-00] | Integration of constraint solving into OMEGA |

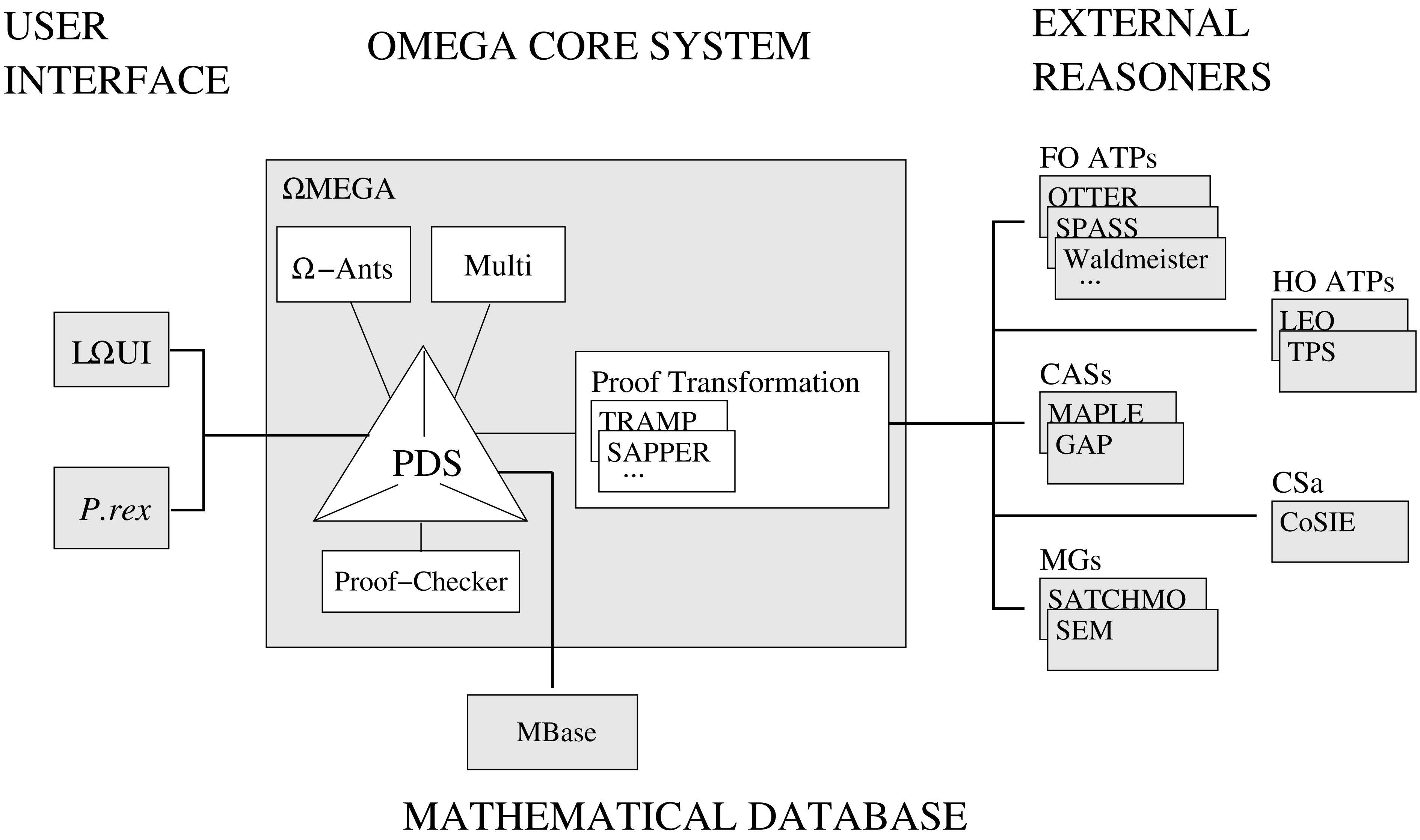
Discussion

+ White-box integration achieved for heterogenous specialist reasoning systems

– Not reached yet: flexible coordination of specialist reasoning systems

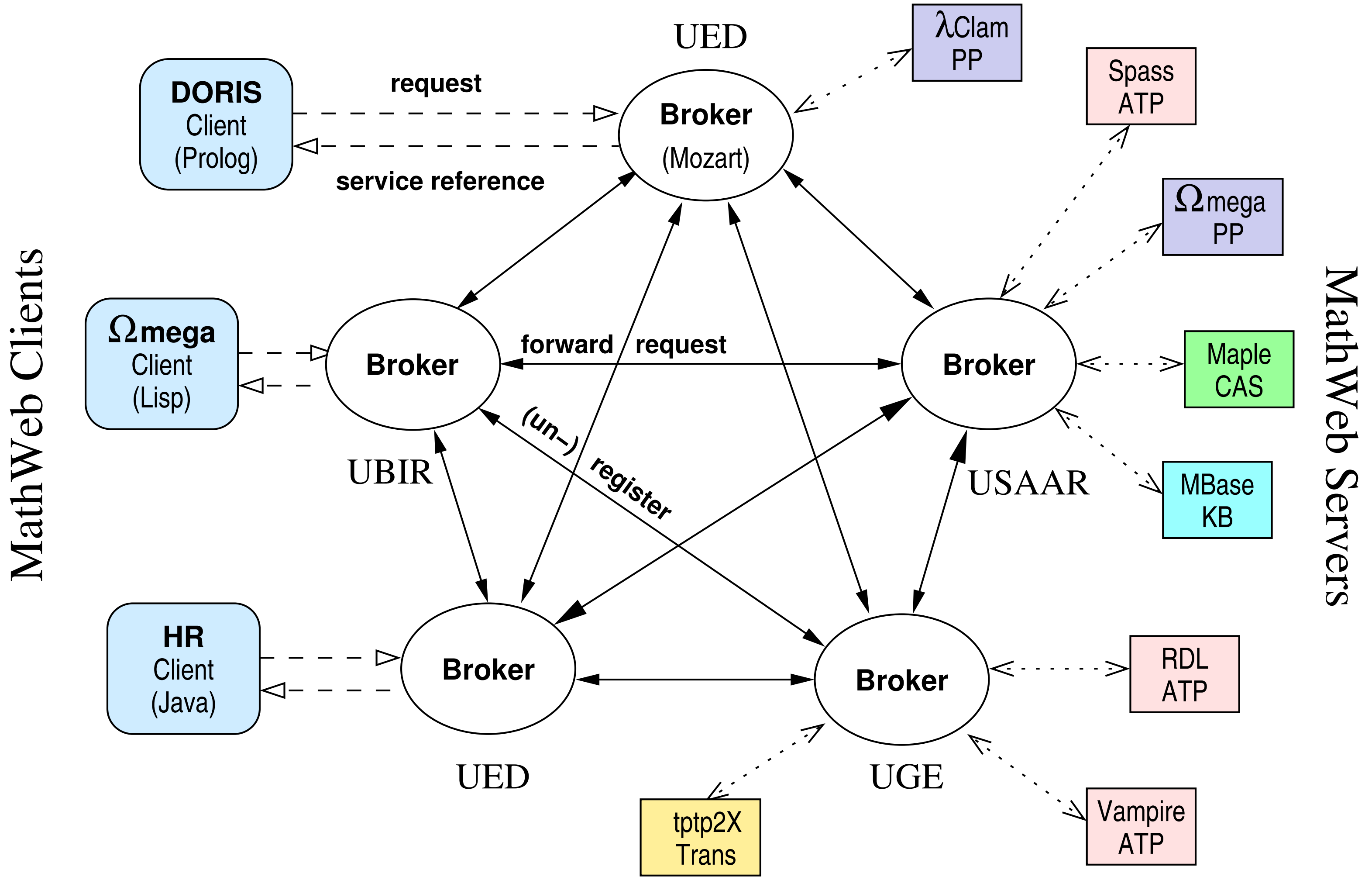
Source: Autexier, Benzmüller

# Modularization



Source: Autexier, Benzmüller

# Modularization



←→ broker to broker communication

◁ – – ▷ client to broker communication (Mozart, XMLRPC, HTTP)

<· · · · > server to broker communication (service offers/requests)

Source: Autexier, Benzmüller

# Modularization

Main References

[KohlhaseZimmer-CADE-02] MathWeb Software Bus

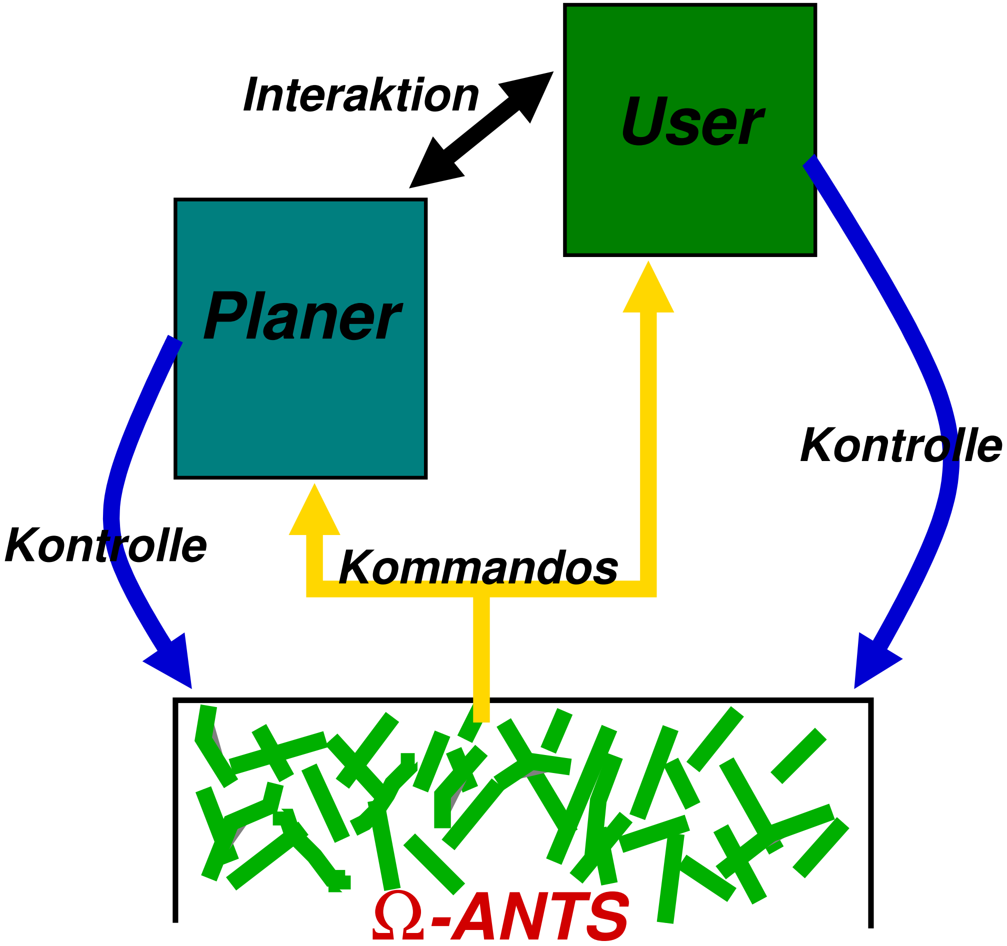
[Kohlhase-AISC-00,Kohlhase-03] OMDoc

Discussion

+ Modular system design supports better maintenance and reuse of system components
+ Better join of resources achieved
– Missing: Intelligent brokering of systems, coordination of systems, . . . , exploitation of and cooperation with QPQ

Source: Autexier, Benzmüller

# Agent-based Theorem Proving

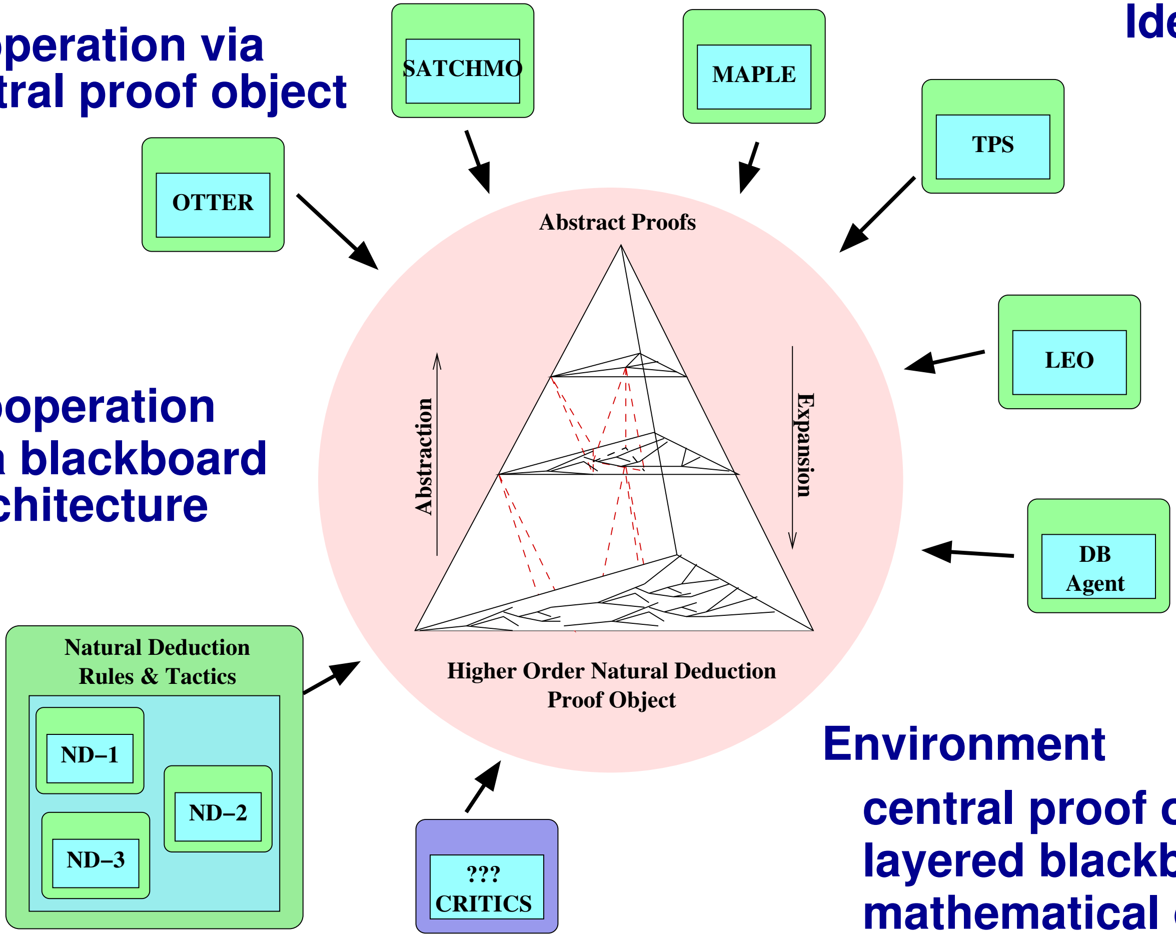


- pro-active support versus passive support
- concurrent versus sequential

Source: Autexier, Benzmüller

# Agent-based Theorem Proving

Cooperation via central proof object



Cooperation via blackboard architecture

ND–Backbone

**Idea**

decentralised control

**Agents**

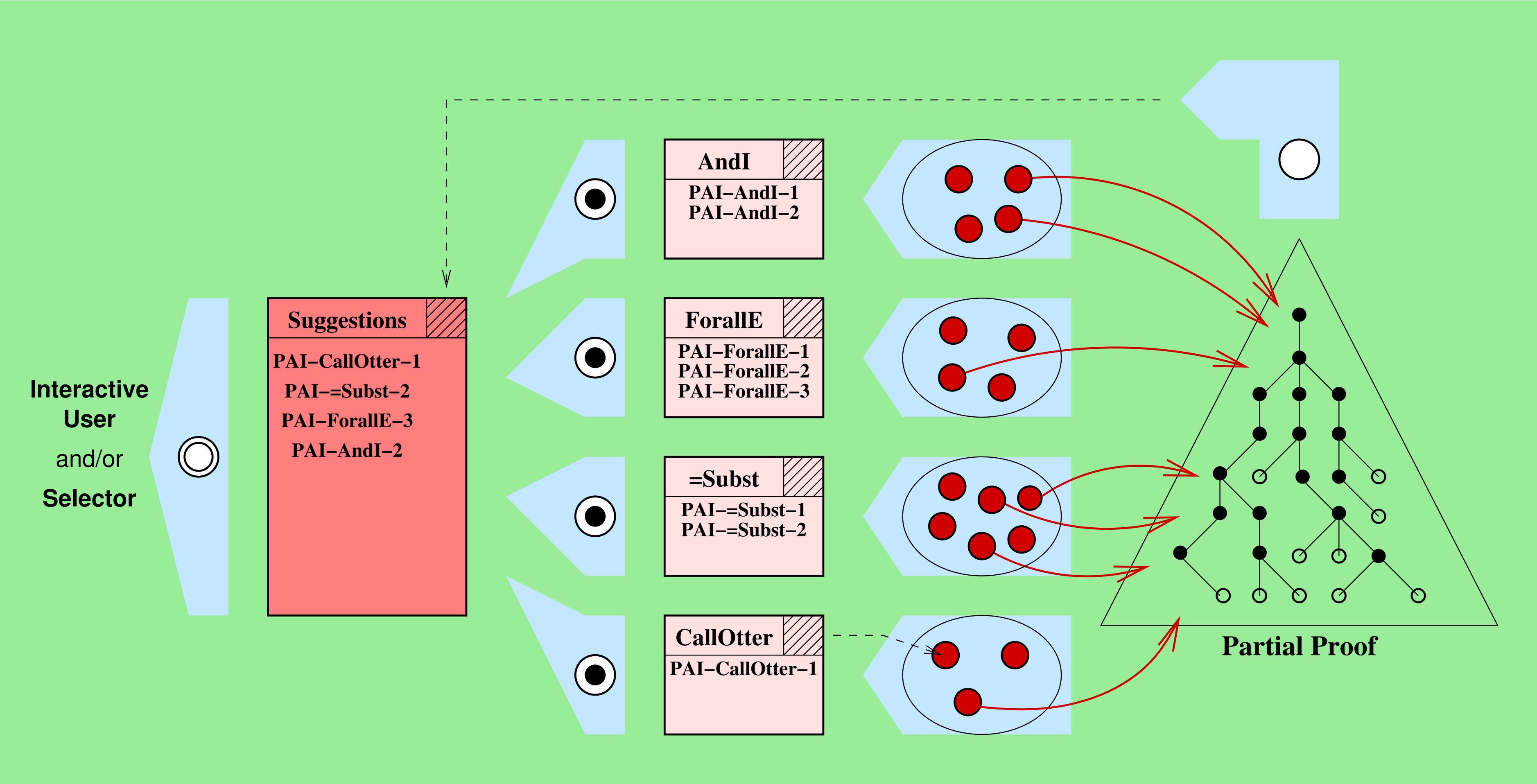
reactive
proactive
heterogenous
simple & complex
cooperative & competetive
distributed via MathWeb
run–time definable
resource adapted

**Environment**

central proof object
layered blackboards (local communication)
mathematical database

Source: Autexier, Benzmüller

# Agent-based Theorem Proving



Source: Autexier, Benzmüller

# Agent-based Theorem Proving

Main References

[BenzmüllerSorge-AIMSA-98, BenzmüllerSorge-EPIA-99, Sorge-PhD-01] OANTS suggestion mechanism

[BenzmüllerSorge-CALCULEMUS-00, BenzmüllerEtAl-KI-01] Agent-based reasoning with external specialist reasoners

[BenzmüllerEtAl-MKM-01] Agent-based search in Knowledge bases

[PolletEtAl] OANTS in interactive proof planning

Discussion

+ Suggestion mechanism useful for interactive theorem proving
+ Looking aside and concurrent search
– Resource-guided agent-based reasoning not fully developed yet

Source: Autexier, Benzmüller

# Novel Research Directions

# Mathematical Assistant In-the-Large

Source: Autexier, Benzmüller

# Current & Future Developments

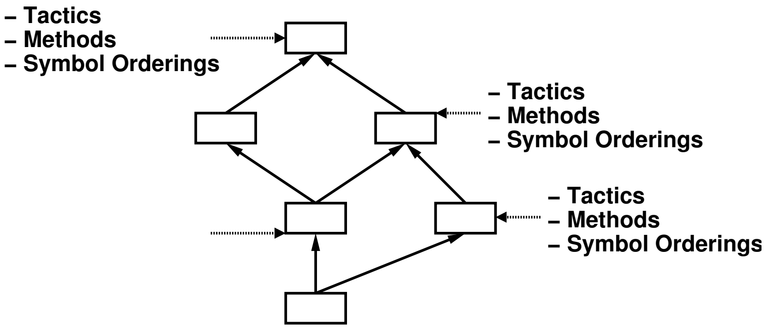
**Theme:** **Towards a smoother integration into spectrum of typical mathematical activities**

- Mathematical Knowledge Management
- Proof development in-the-large
  - Lifting the level of proof construction
  - Combination/Integration of proof search paradigms
  - Integration of structured mathematical knowledge
- Towards typical mathematical activities
  - Writing mathematical publications
  - Tutoring for mathematics students

Source: Autexier, Benzmüller

# Mathematical Knowledge Management

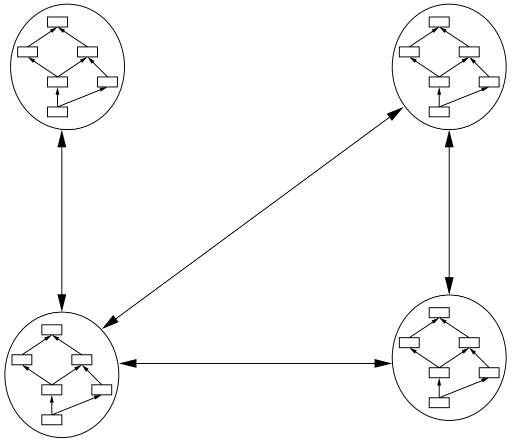
1. Types of knowledge
   - Formalized mathematical theories
   - Structured
   - Domain specific proof knowledge tactics, proof-planning methods, symbol orderings, . . .



Source: Autexier, Benzmüller

# Mathematical Knowledge Management

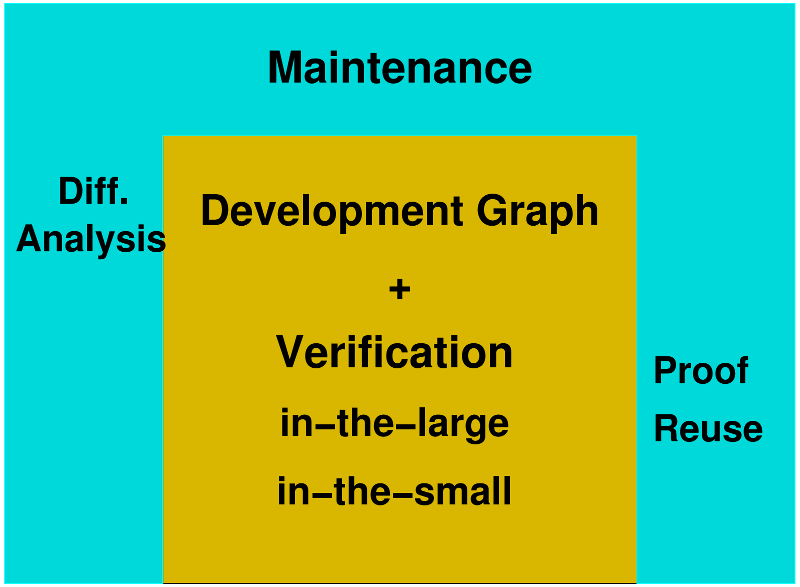
1. Types of knowledge
2. Distributed over different physical locations
   - Origin tracking, remote access, . . .



Source: Autexier, Benzmüller

# Mathematical Knowledge Management

1. Types of knowledge
2. Distributed over different physical locations
3. Evolution of mathematical knowledge
   - Management of change
     Benefit from experience with MAYA
   - Versioning



Maintenance

Diff. Analysis

Development Graph

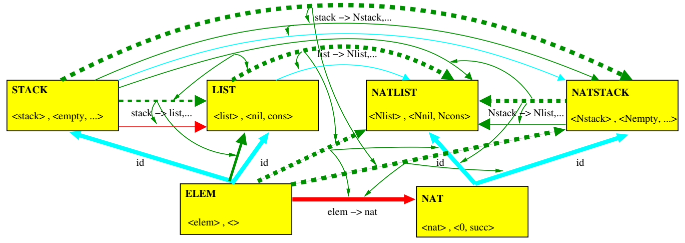
+

Verification

in–the–large

in–the–small

Proof Reuse



Source: Autexier, Benzmüller

# In-the-large Proof Development

1. **Lifting the level of proof construction**
   - Support proof development directly on the Assertion Level
   - CoRe-proof calculus [PhD-Autexier-03]
     - Supports determination of assertions for subformulas
     - Supports application of assertion to subformulas
     - New logic engine for ΩMEGA

Source: Autexier, Benzmüller

# In-the-large Proof Development

1. **Lifting the level of proof construction**
2. **Combination/Integration of proof search paradigms**

- Procedural Tactics, declarative proof-planning, distributed ΩANTS

  Develop heterogenous paradigm [AutexierBenmuellerHutter-SEKI-03]
- All work on the new proof calculus provided by CoRE
- Common, paradigm-independent proof object eases combination
- Adaptation of ΩANTS to new interface [MsC-Thesis-Hübner]

Source: Autexier, Benzmüller

# In-the-large Proof Development

1. **Lifting the level of proof construction**
2. **Combination/Integration of proof search paradigms**
3. **Integration of structured mathematical knowledge**
   - Search for appropriate assertions in structured mathematical theories [Vo-Autexier-Benzmüller-IJCAI-03]
   - Redesign of MATHWEB-SB [PhD J. Zimmer]
     - Accommodate existing Multi-Agent-System description and communication standards
     - Integrate automated problem solving capabilities

Source: Autexier, Benzmüller

# Supporting Mathematical Publications

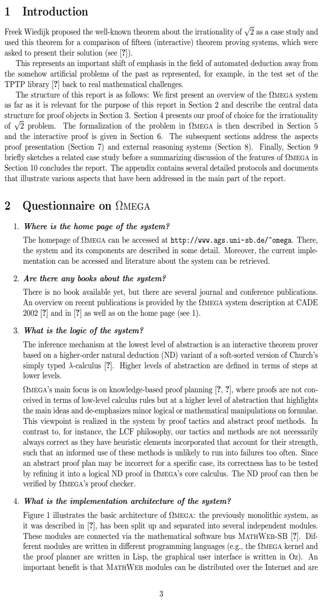
- Writing mathematical papers in publishable format
- Relate parts in paper to formally defined objects in MBASE (theories, symbols, definitions, lemmas, proofs)
  - Initialize paper wrt. background theory in MBASE
  - Writing definitions and lemmas gives (automatically) rise to formal counter-parts in MBASE
  - Written proofs give rise to formal proof objects in ΩMEGA
- Vision: Certified mathematical publications

Source: Autexier, Benzmüller

# System Architecture

Provision of Semantic Background

## TeXmacs



### 1 Introduction

Freek Wiedijk proposed the well-known theorem about the irrationality of $\sqrt{2}$ as a case study and used this theorem for a comparison of fifteen (interactive) theorem proving systems, which were asked to present their solution (see [?]).

This represents an important shift of emphasis in the field of automated deduction away from the somehow artificial problems of the past as represented, for example, in the test set of the TPTP library [?] back to real mathematical challenges.

The structure of this report is as follows: We first present an overview of the ΩMEGA system as far as it is relevant for the purpose of this report in Section 2 and describe the central data structure for proof objects in Section 3. Section 4 presents our proof of choice for the irrationality of $\sqrt{2}$ problem. The formalization of the problem in ΩMEGA is then described in Section 5 and the interactive proof is given in Section 6. The subsequent sections address the aspects proof presentation (Section 7) and external reasoning systems (Section 8). Finally, Section 9 briefly sketches a related case study before a summarizing discussion of the features of ΩMEGA in Section 10 concludes the report. The appendix contains several detailed protocols and documents that illustrate various aspects that have been addressed in the main part of the report.

### 2 Questionnaire on ΩMEGA

1. ***Where is the home page of the system?***

   The homepage of ΩMEGA can be accessed at `http://www.ags.uni-sb.de/~omega`. There, the system and its components are described in some detail. Moreover, the current implementation can be accessed and literature about the system can be retrieved.

2. ***Are there any books about the system?***

   There is no book available yet, but there are several journal and conference publications. An overview on recent publications is provided by the ΩMEGA system description at CADE 2002 [?] and in [?] as well as on the home page (see 1).

3. ***What is the logic of the system?***

   The inference mechanism at the lowest level of abstraction is an interactive theorem prover based on a higher-order natural deduction (ND) variant of a soft-sorted version of Church's simply typed $\lambda$-calculus [?]. Higher levels of abstraction are defined in terms of steps at lower levels.

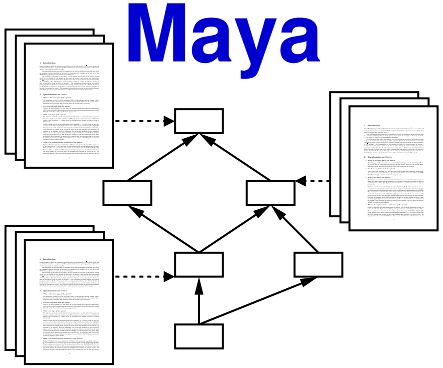
   ΩMEGA's main focus is on knowledge-based proof planning [?, ?], where proofs are not conceived in terms of low-level calculus rules but at a higher level of abstraction that highlights the main ideas and de-emphasizes minor logical or mathematical manipulations on formulae. This viewpoint is realized in the system by proof tactics and abstract proof methods. In contrast to, for instance, the LCF philosophy, our tactics and methods are not necessarily always correct as they have heuristic elements incorporated that account for their strength, such that an informed use of these methods is unlikely to run into failures too often. Since an abstract proof plan may be incorrect for a specific case, its correctness has to be tested by refining it into a logical ND proof in ΩMEGA's core calculus. The ND proof can then be verified by ΩMEGA's proof checker.

4. ***What is the implementation architecture of the system?***

   Figure 1 illustrates the basic architecture of ΩMEGA: the previously monolithic system, as it was described in [?], has been split up and separated into several independent modules. These modules are connected via the mathematical software bus MATHWEB-SB [?]. Different modules are written in different programming languages (e.g., the ΩMEGA kernel and the proof planner are written in Lisp, the graphical user interface is written in Oz). An important benefit is that MATHWEB modules can be distributed over the Internet and are

3

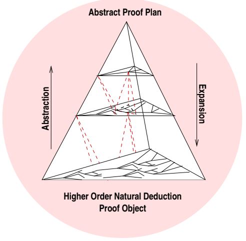
## MBase/ Maya



Storing Documents

Interactive Proof Support

## Omega



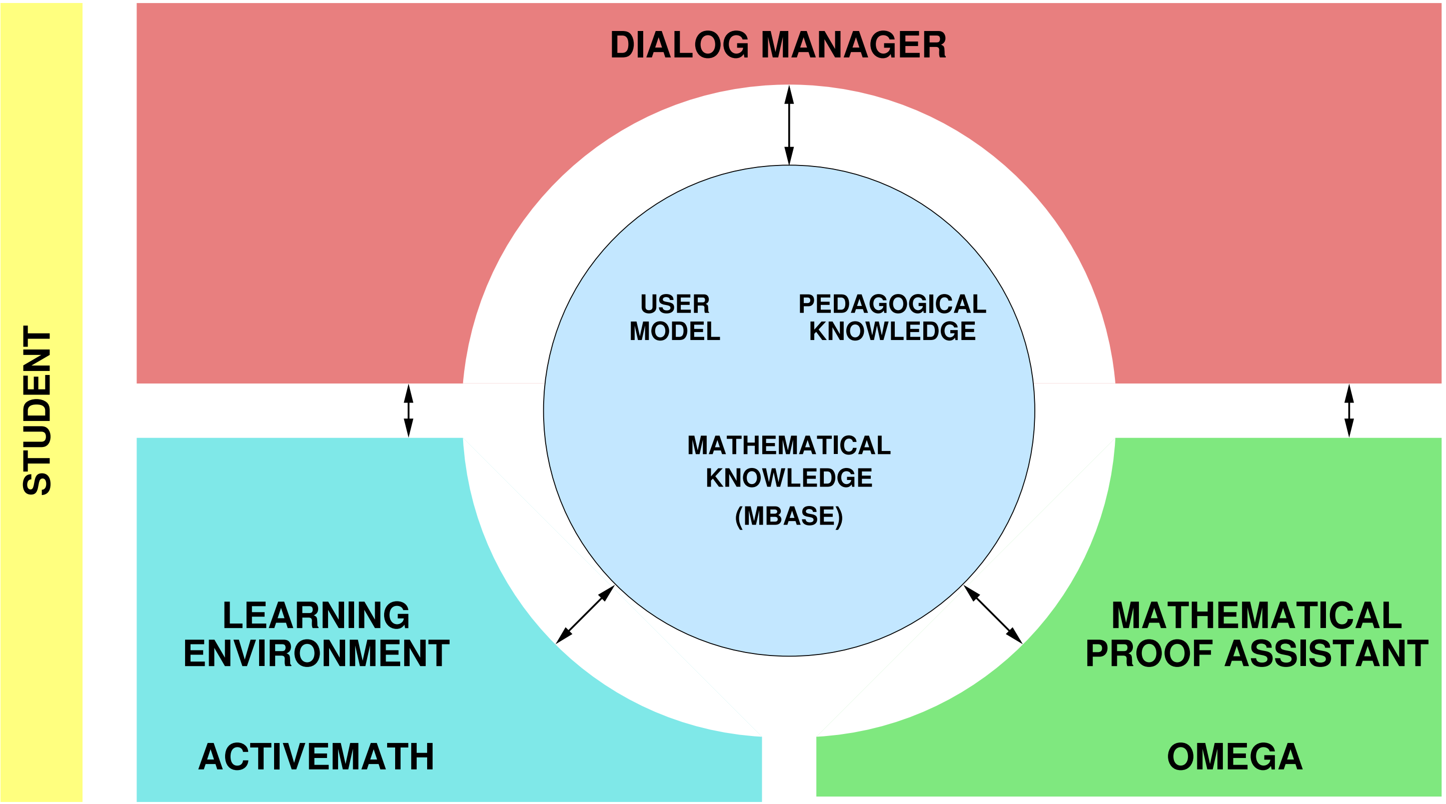
Source: Autexier, Benzmüller

# Supporting Mathematical Publications

- Scenario: Students develop proofs in a natural language dialog and are advised by the system
- Linguistic analysis of student utterances
- Reconstruction of probable proof
- Comparison to tutor proof results in advise for student

Source: Autexier, Benzmüller

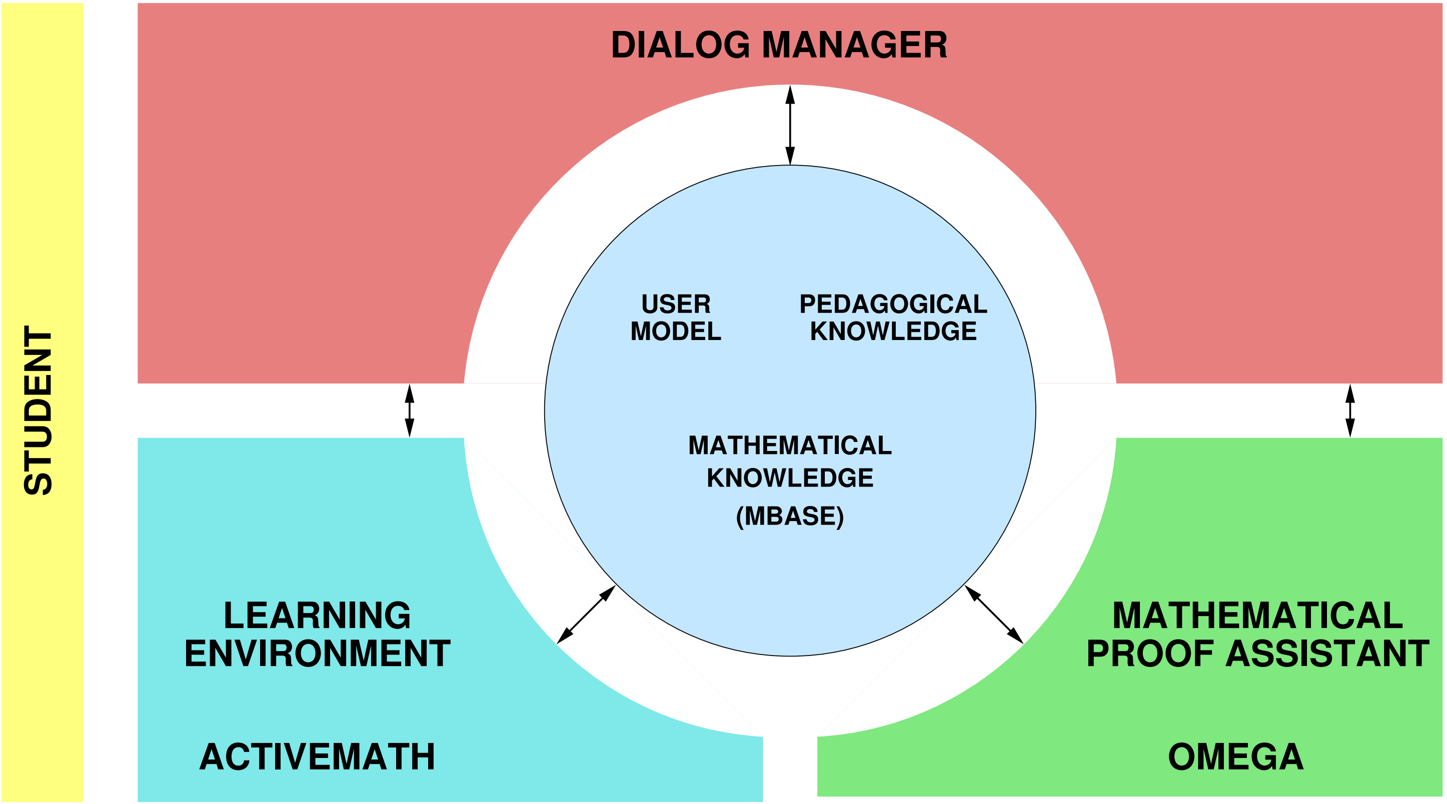
# DIALOG-System Architecture



Source: Autexier, Benzmüller

# DIALOG-System Architecture

ΩMEGA



- Activemath ⇒ Talk E. Melis
- Underspecification of Proofs ⇒ Talk A. Fiedler

Source: Autexier, Benzmüller

UNIVERSITÄT DES SAARLANDES

# Lessons learned...

- Modularization was important for ΩMEGA-system and -research group
  - work on clear interfaces and interface/communication languages OMDOC
  - eases reuse and join of resources MBASE, MATHWEB-SB
- Don't fight over proof search paradigms
  Concentrate on joining strengths of each to finally build a MA
- System-stability would highly benefit from
  - having long-term employed software engineer (Funding problem)
  - applying high-qualitty software development principles
- System development and stability depends on teamwork spirit

Source: Autexier, Benzmüller