ΩMEGA

A Mathematics Assistance System

and

DIALOG: Natural Language-based Interaction with a Mathematics Assistance System

Serge Autexier and Christoph Benzmüller and Martin Pollet

Saarland University, Saarbrücken, Germany

October 20th 2004

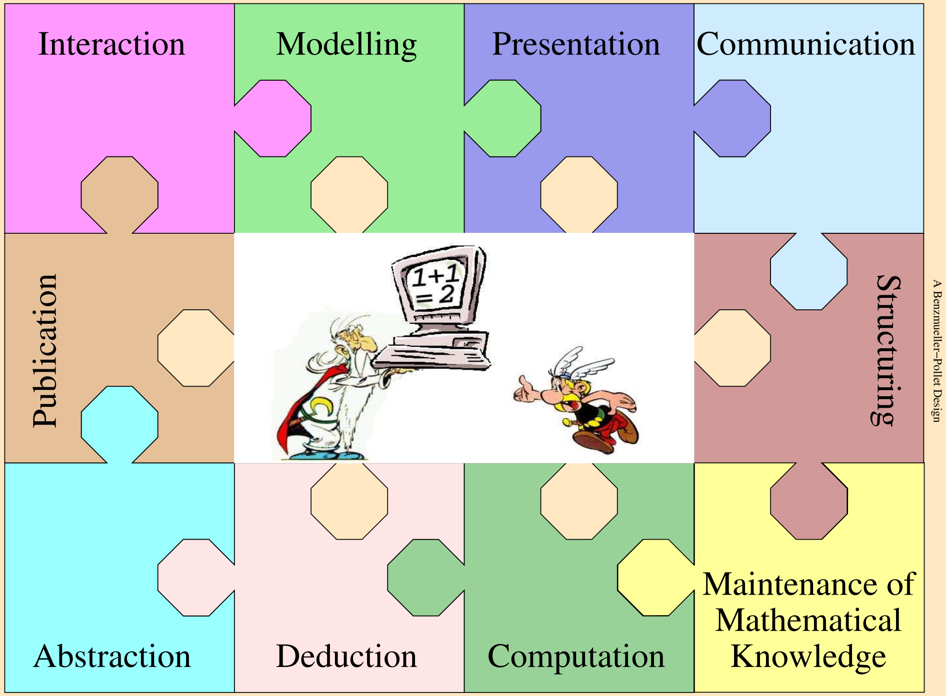
Saarbrücken, Germany

Overview

- Mathematics Assistance Environments
- The ΩMEGA Project
  - Mathematics Assistant In-the-small
    – research directions since early 90s –
  - Mathematics Assistant In-the-large
    – novel research directions –
- DIALOG: Natural Language-based Interaction with a Mathematics Assistance System

Mathematics Assistance Systems

Integrated Mathematics Assistance Environment vs. 'Pen-and-Paper' Mathematics



Join of ressources necessary

Applications

Mathematics research
Mathematics education
Formal methods

System level: Coq, NuPrl, Isabelle/HOL, PVS, Theorema, ΩMEGA, CLam, ...

Research Networks: Calculemus, MKM, Monet, MoWGLI, ...

Source: Autexier & Benzmüller & Pollet

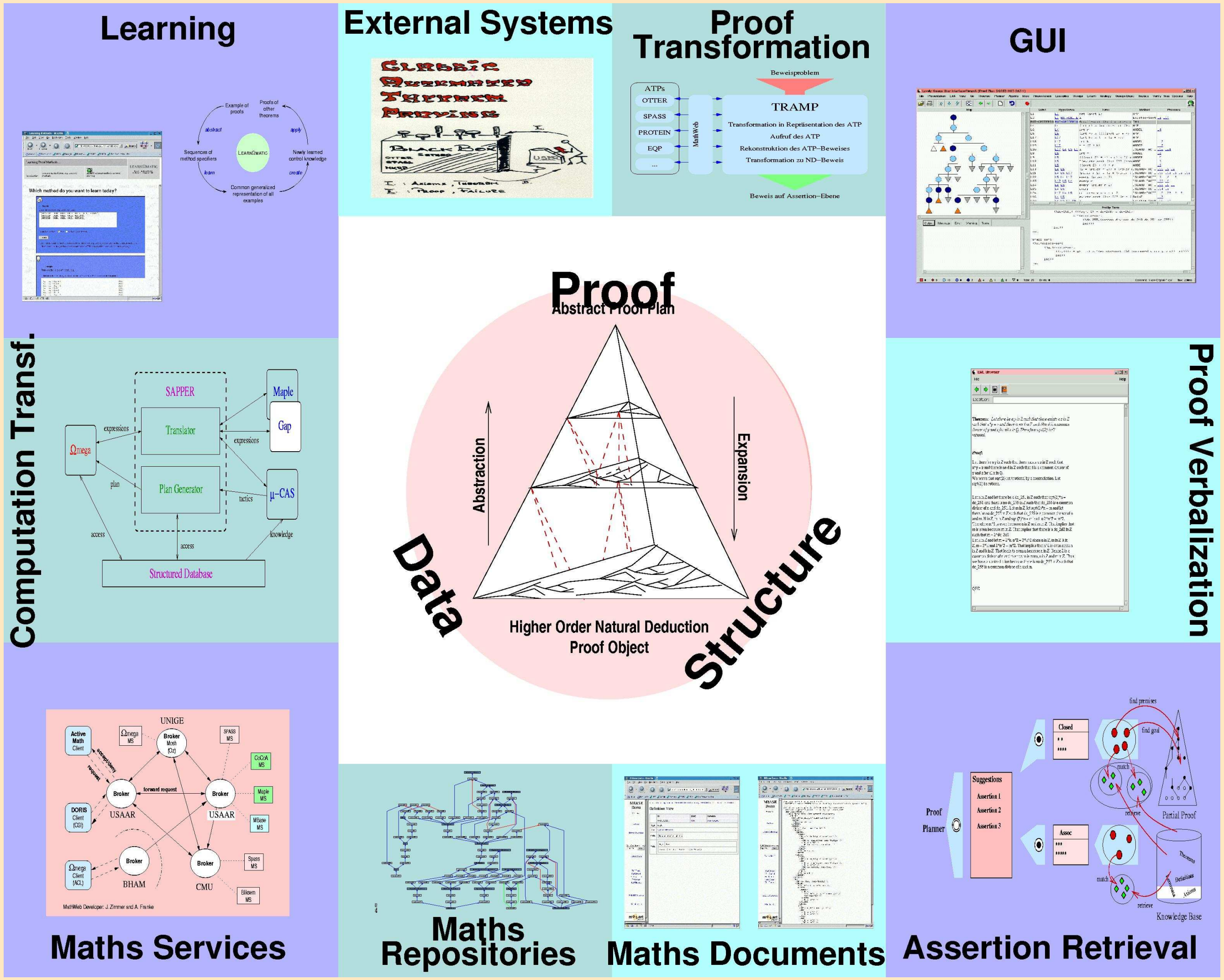
OMEGA GROUP

Mathematics Assistance System

In-the-small

Research directions in the ΩMEGA project since the early 90s

The ΩMEGA Instance



Source: Autexier & Benzmüller & Pollet

MBASE & OMDOC

MBASE: mathematical knowledge base

- universal syntax for mathematical documents: OMDOC
- mathematical texts in varying degree of formalisation
- query language for OPENMATH formulas

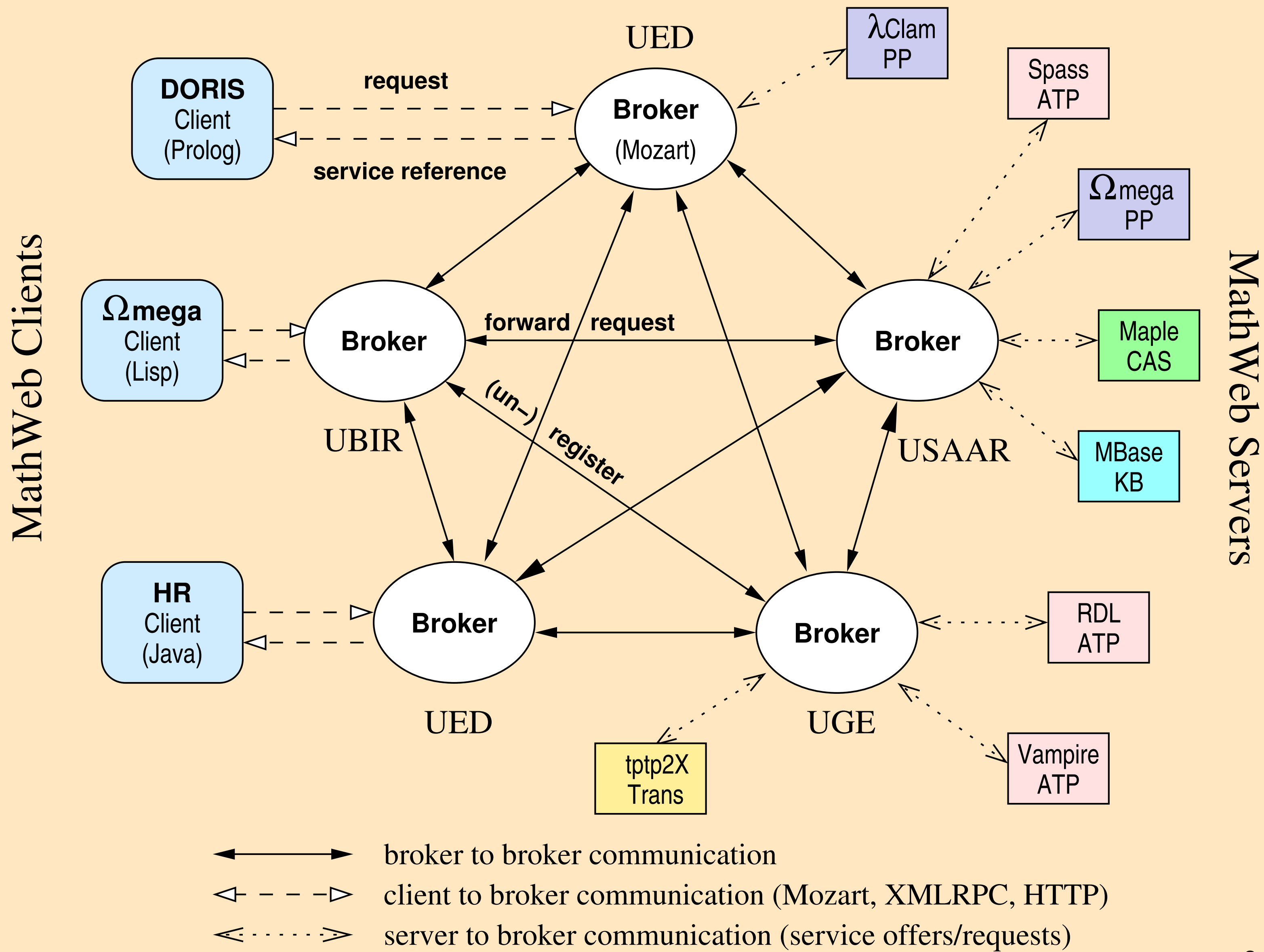
Discussion

+ First step towards system independence

– Version control: concurrent, joint development of mathematical knowledge

– System independent representation formats for proof rules, tactics, methods, and control knowledge

External Specialist Reasoners

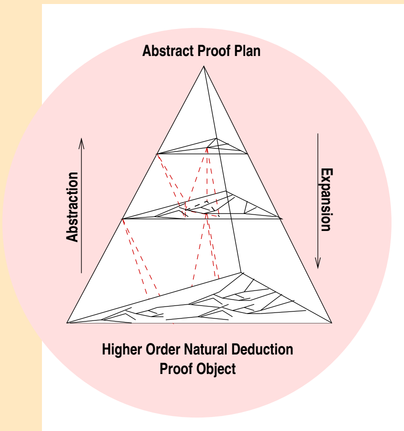


Source: Autexier & Benzmüller & Pollet

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 Tools

Proof Transformation
Proof Reconstruction

problem' →
← result(cryptic)

External Specialist Reasoner

available for:

FOL ATPs (TRAMP), CASs (SAPPER), TPS, constraint solving

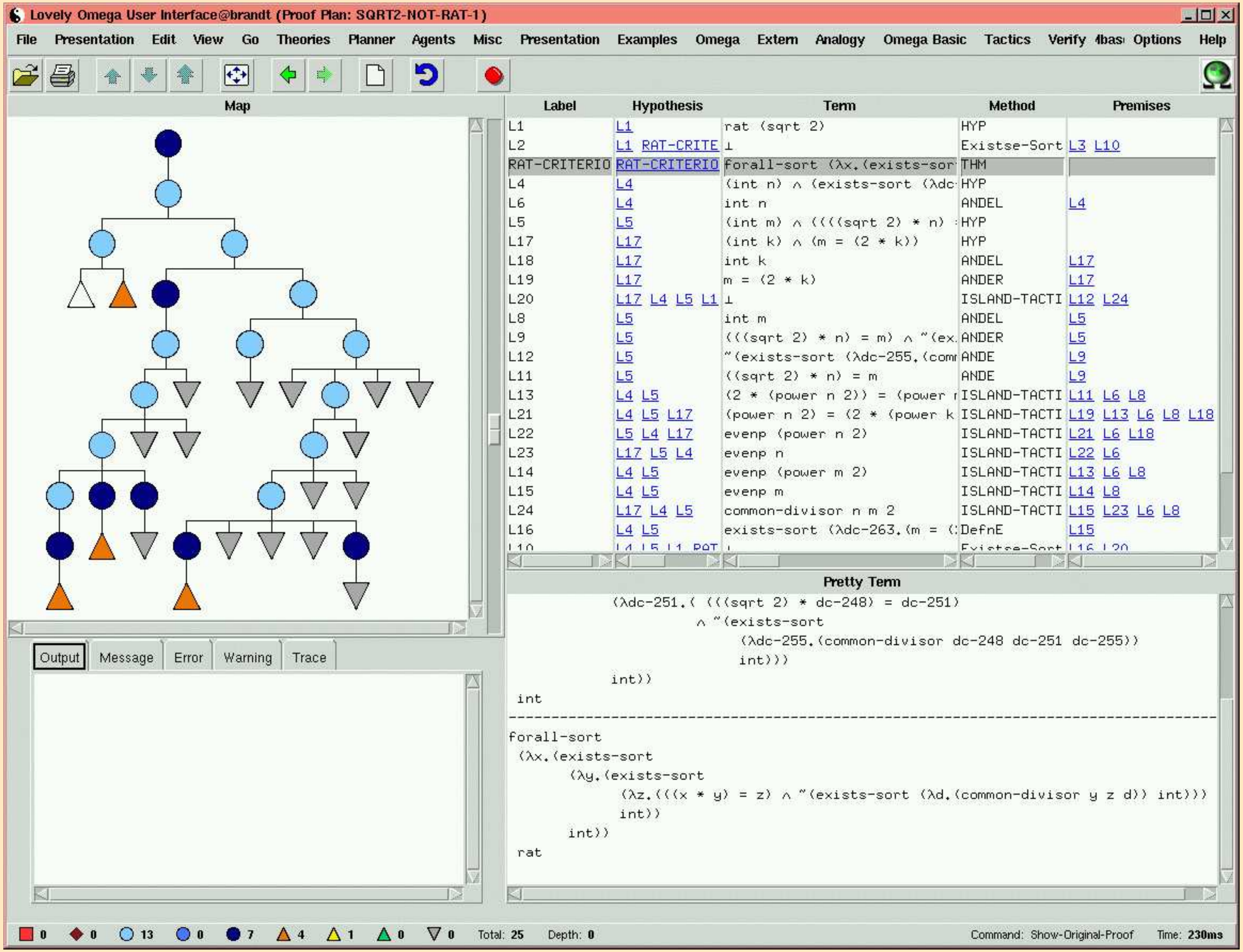
Source: Autexier & Benzmüller & Pollet

External Specialist Reasoners

Discussion

+ White-box integration achieved for heterogenous specialist reasoning systems
+ Modular system design supports better maintenance and reuse of system components
+ Better join of resources achieved
– Not reached yet: flexible coordination of specialist reasoning systems
– Missing: Intelligent brokering of systems, coordination of systems, . . . , exploitation of and cooperation with QPQ

User Interface



Source: Autexier & Benzmüller & Pollet

User Interface

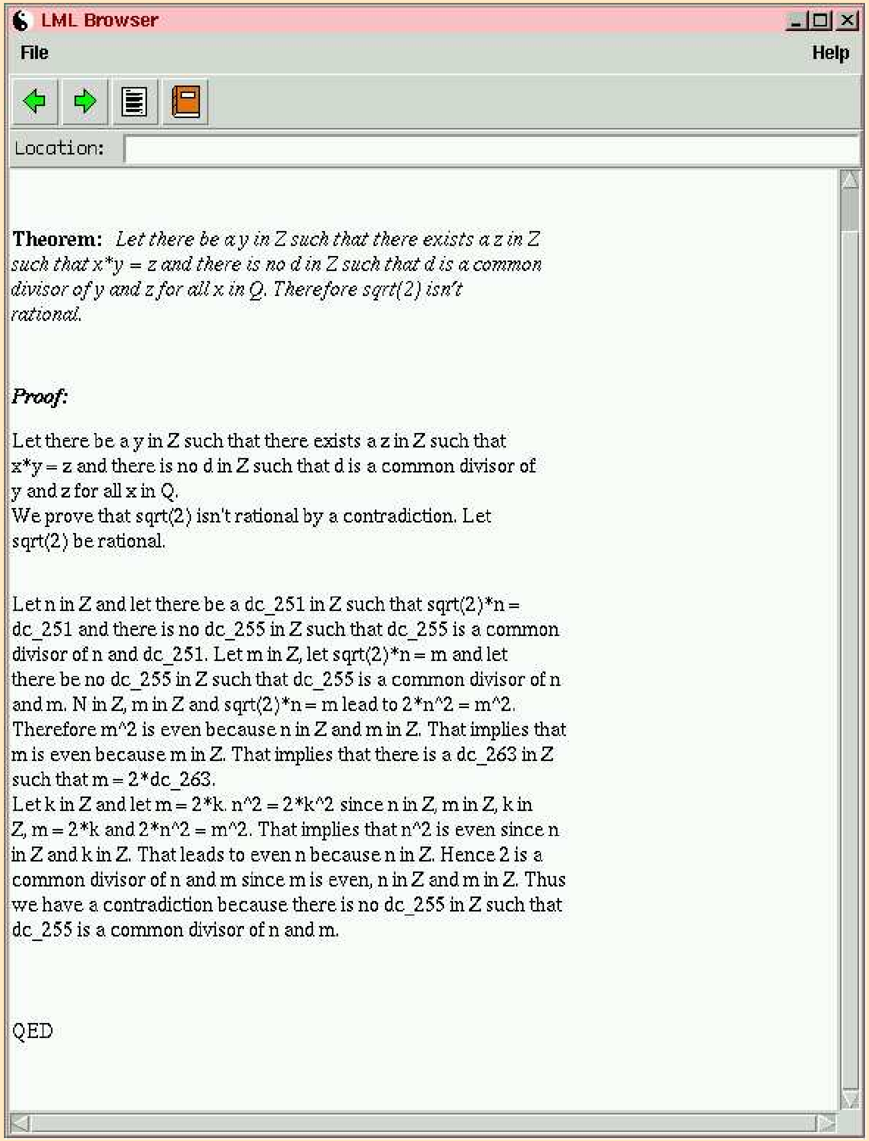
LOUI: Lovely OMEGA User Interface

+ Support for different views on proof developments: linearized ND style, proof tree, natural language
+ Navigation through different levels of the PDS
+ Support for interactive proof construction

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

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

Proof Verbalization



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 & Pollet

Proof Verbalization

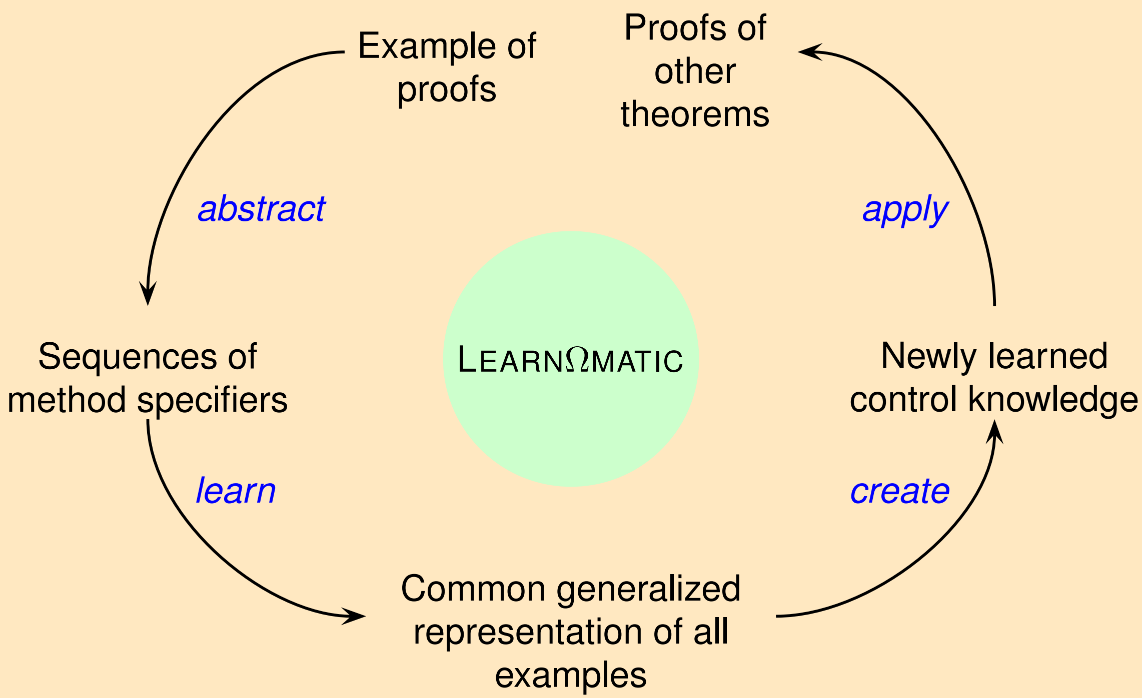
Discussion

+ Flexible, adaptable, non-template based proof verbalization

– Missing: full natural language for tutorial dialogs at assertion level

LEARNΩMATIC

Hybrid system consisting of learn engine and deduction system.

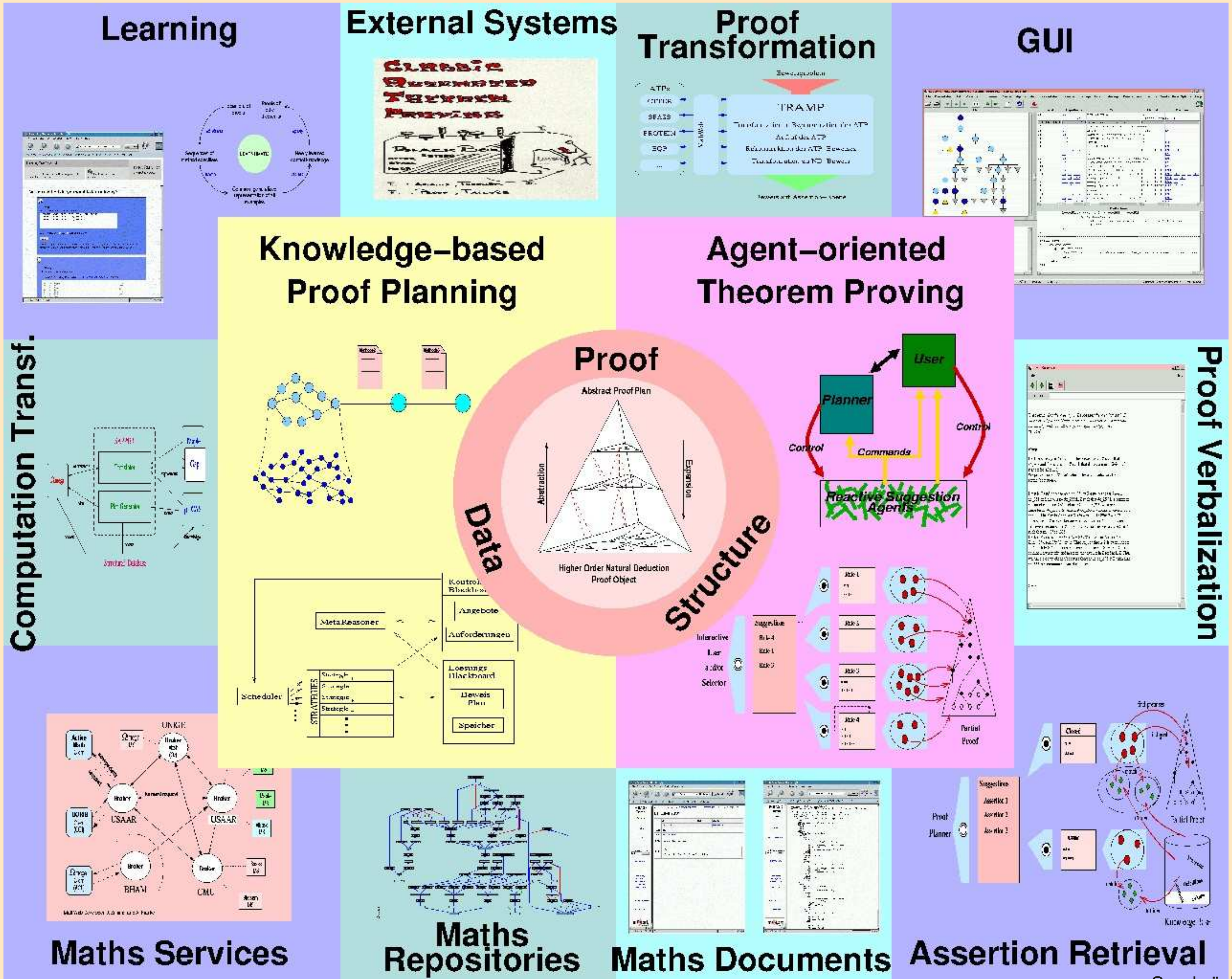


+ More theorems provable, proof search more directed, and shorter proofs

– What other information can be considered for learning?

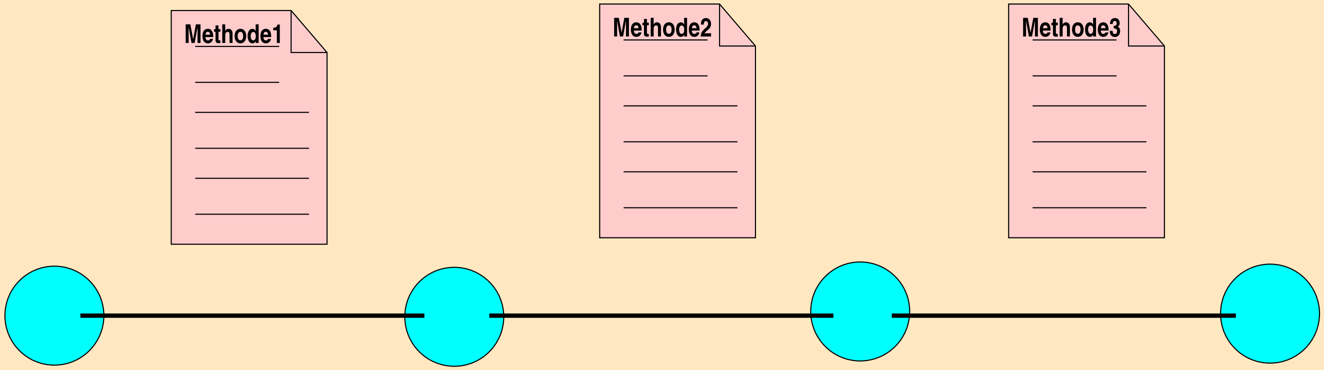
Source: Autexier & Benzmüller & Pollet

The Complete Picture



Source: Autexier & Benzmüller & Pollet

Proof Planning



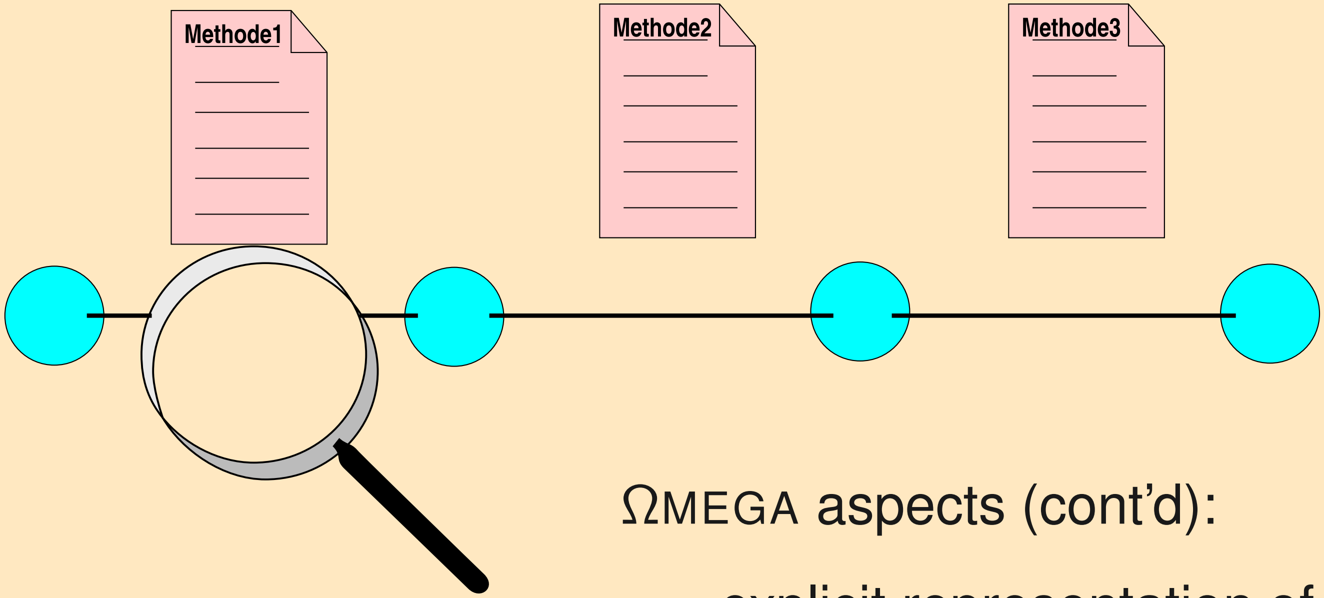
ΩMEGA born in early 90s; inspired by [Bundy88]

- 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

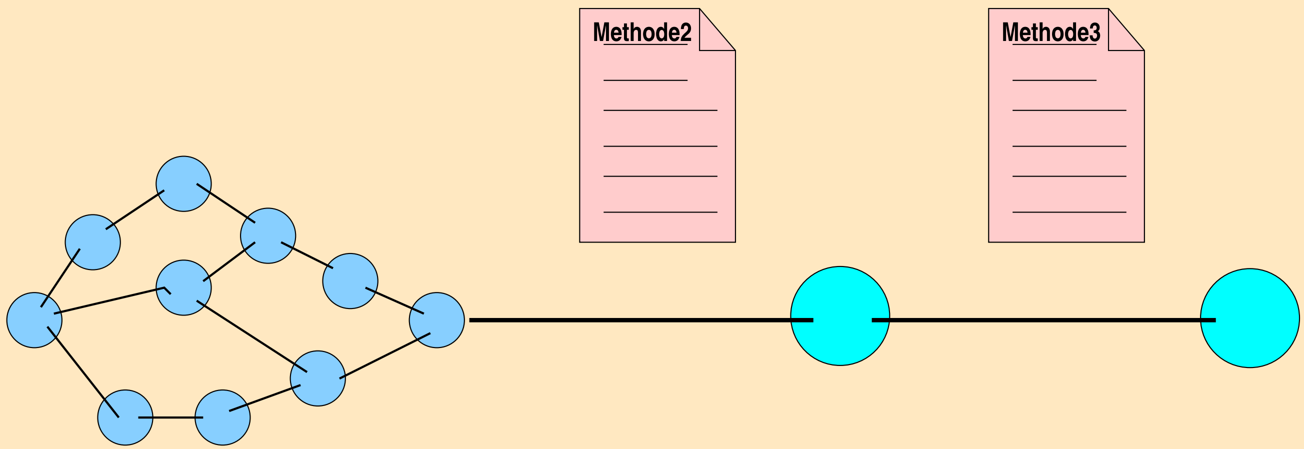
Proof Planning



ΩMEGA aspects (cont'd):

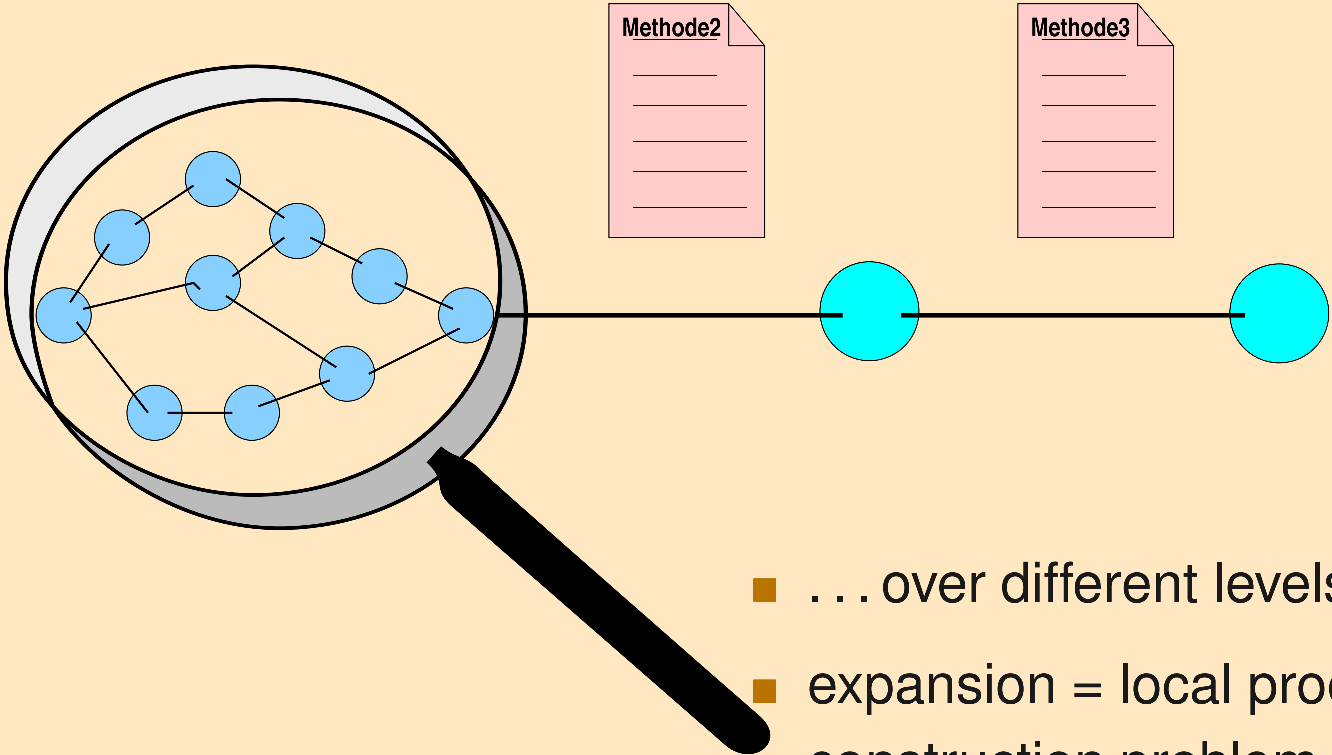
- explicit representation of proof plans
- under-specification of pre-conditions: potentially non-sound proof plans
- soundness guaranteed via ...

Proof Planning



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

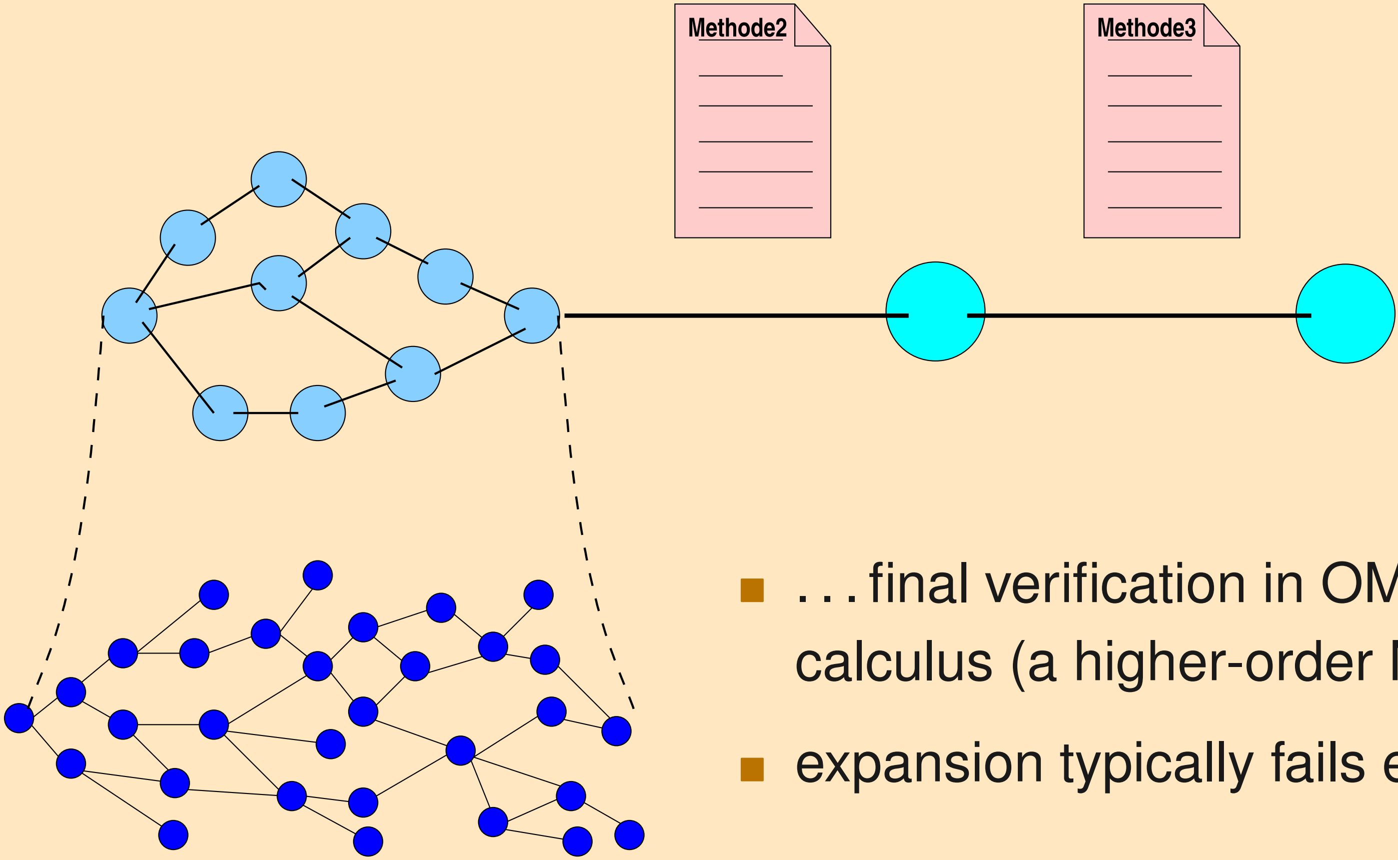
Proof Planning



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

Source: Autexier & Benzmüller & Pollet

Proof Planning



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

Proof Planning

Case studies

ε-δ-proofs: Use of constraint-solver and computer algebra system

Residue class properties: combination of CASs and theory formation system HR, classification of $\sim$18.000 structures

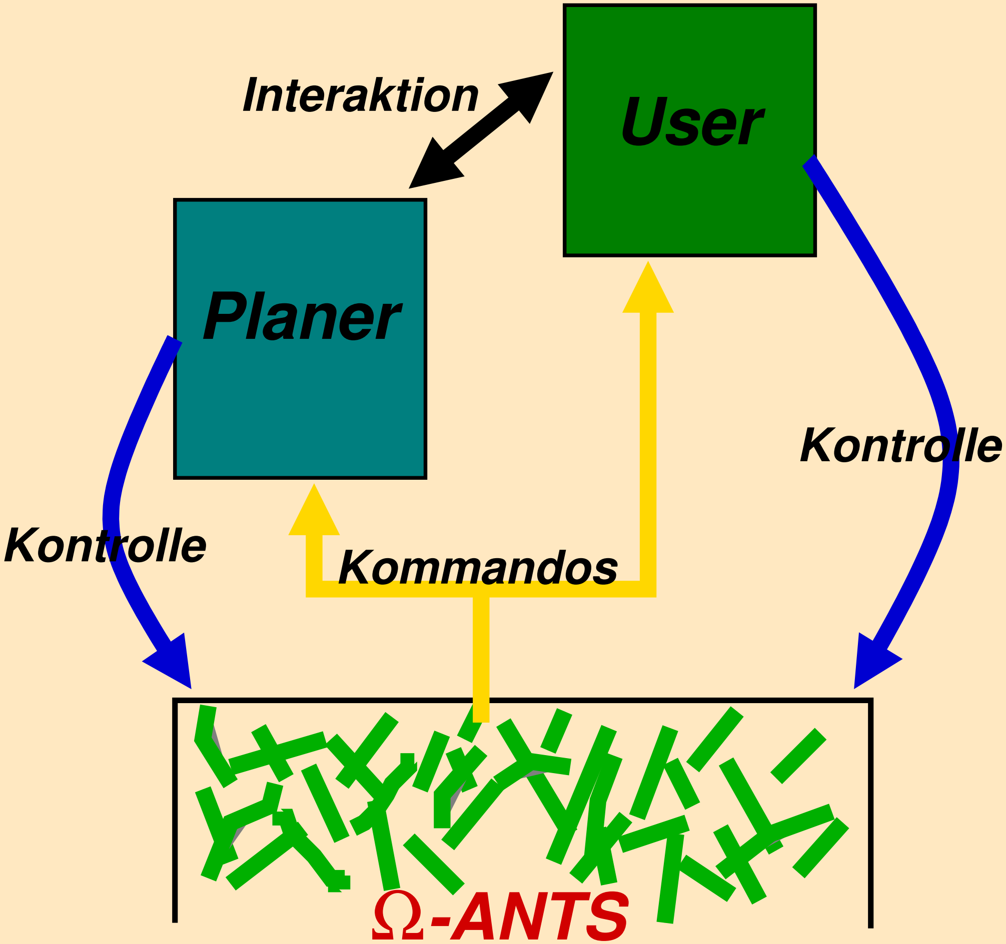
Verification of GAP computations on permutation groups: Verification by proof search instead of hard-wired scripts

Discussion

+ Automatic proofs for problem classes in specific domains
+ Coordination of systems
– Brittleness and logic layer dependency

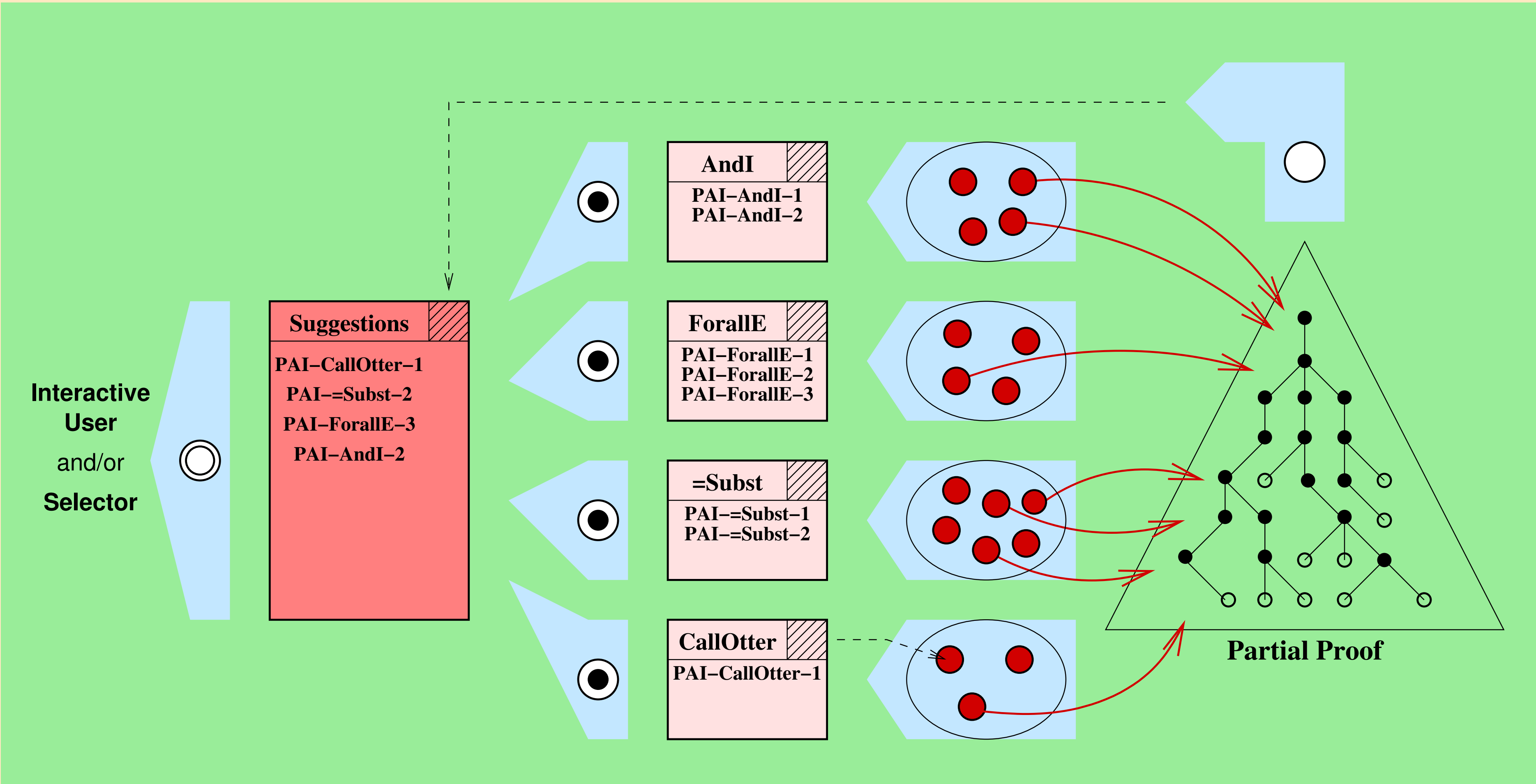
Source: Autexier & Benzmüller & Pollet

Agent-based Theorem Proving



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

Agent-based Theorem Proving

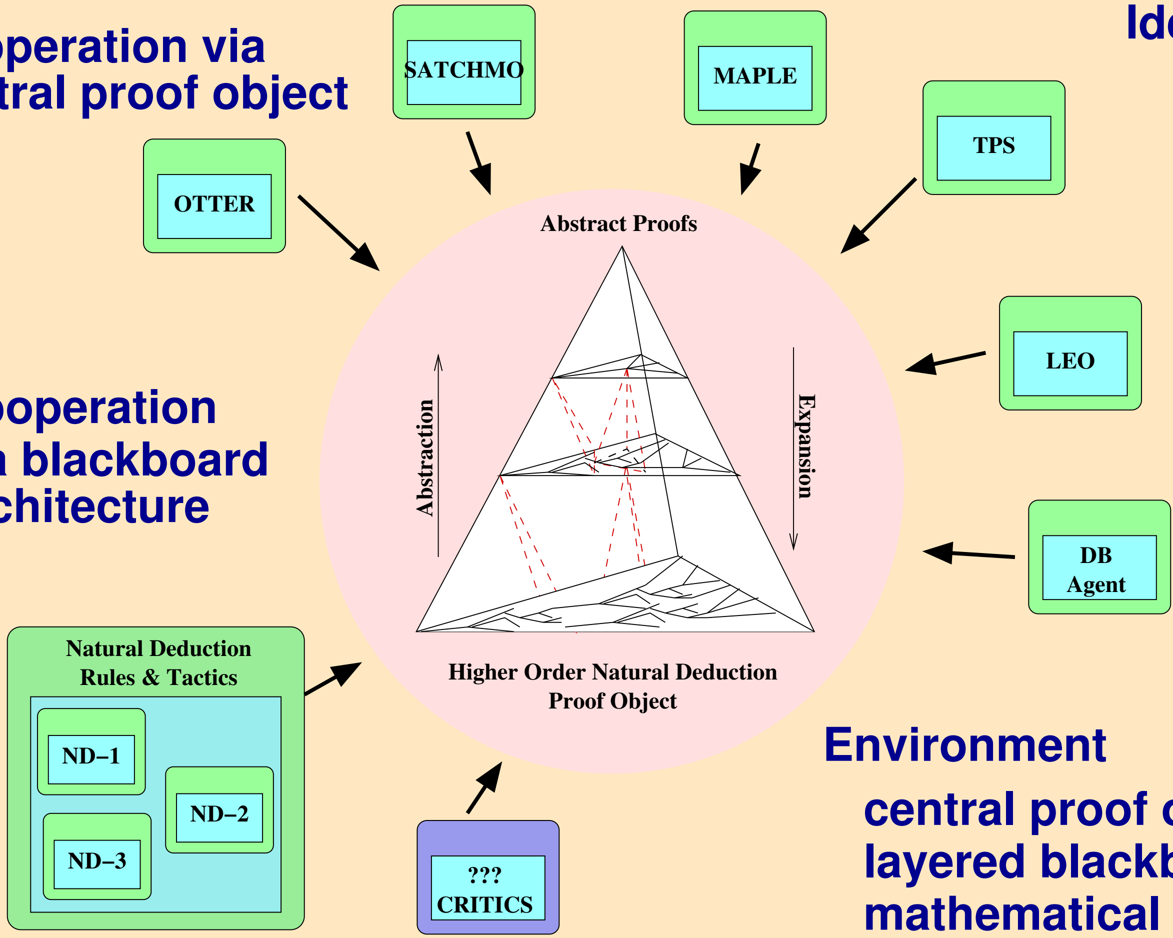


Source: Autexier & Benzmüller & Pollet

Agent-based Theorem Proving

Cooperation via central proof object

Idea
decentralised control



Cooperation via blackboard architecture

ND−Backbone

Agents
- reactive
- proactive
- heterogenous
- simple & complex
- cooperative & compete
- distributed via MathWe
- run−time definable
- resource adapted

Environment
- central proof object
- layered blackboards (local communication)
- mathematical database

Source: Autexier & Benzmüller & Pollet

Agent-based Theorem Proving

Applications

- Tactic suggestion mechanism & automated agent-based proving
- Reasoning with external specialist reasoners
- Agent-based search in knowledge bases
- 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 & Pollet

Novel Research Directions

Mathematics Assistance System In-the-Large

Current & Future Developments

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

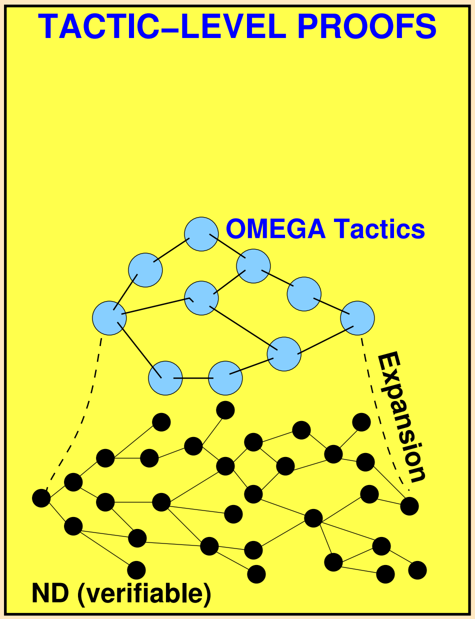
- Proof development in-the-large
  - Lifting the level of proof construction
  - Combination/Integration of proof search paradigms
  - Integration of structured mathematical knowledge
- Mathematical Knowledge Management
- Towards typical mathematical activities
  - Writing mathematical publications
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

Assertion Level Reasoning

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

Proof. *(by contradiction)*

Assume $\sqrt{2}$ is rational, that is, there exist natural numbers m, n *with no common divisor such that* $\sqrt{2} = \frac{\mathrm{m}}{\mathrm{n}}$. *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* $m = 2k$. *...*



Result:

Tactic-based theorem provers

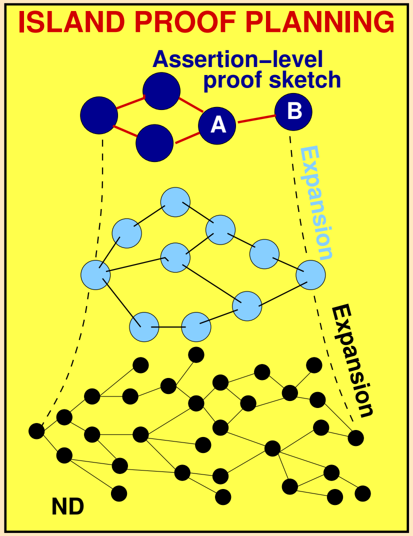
like PVS, ISABELLE, COQ, etc.

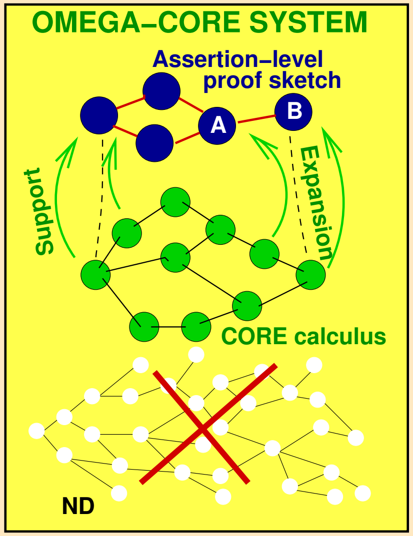
(and OMEGA) can construct a proof –

but not at an adequate level

Source: Autexier & Benzmüller & Pollet

Proof Planning & The CoRe Calculus





- new explicit layer for proof sketches
- easy to add to proof data structure
- proof sketches may be unsound
- verification by expansion to ND

Problem:

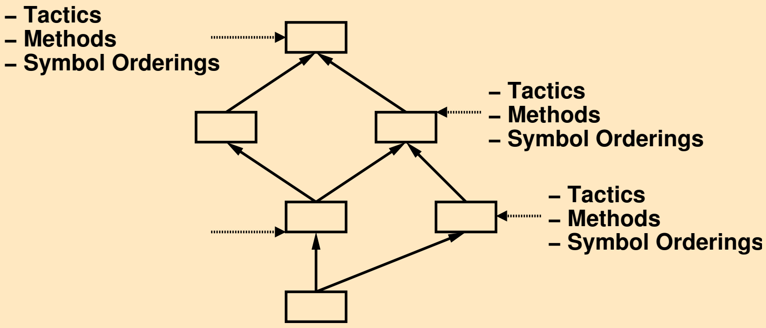
- expansion distance to base-layer

- CORE is more natural/abstract-level base calculus for proof assistants
- subsumes old ND base and (parts of) the old tactic layer
- provides more direct, constructive support for upper layer(s)

Source: Autexier & Benzmüller & Pollet

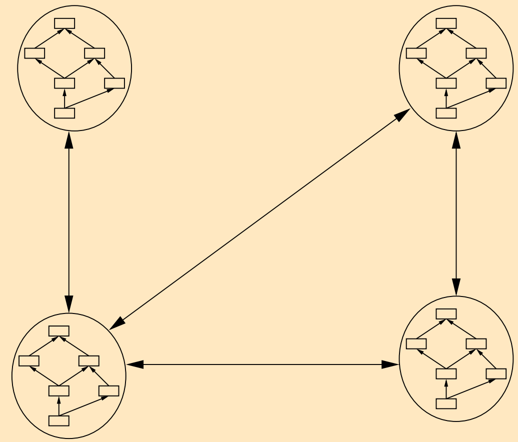
Mathemat. Knowledge Management

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



Mathemat. Knowledge Management

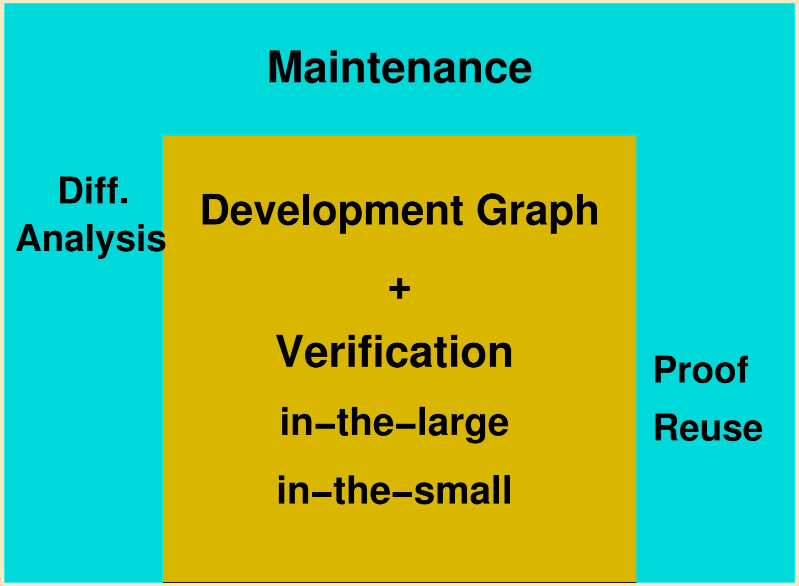
1. Types of knowledge
2. Distributed over different physical locations
   - Origin tracking, remote access, . . .

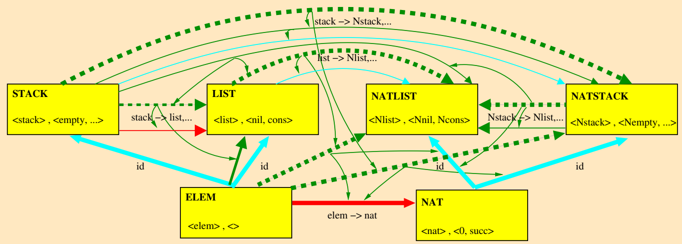


Source: Autexier & Benzmüller & Pollet

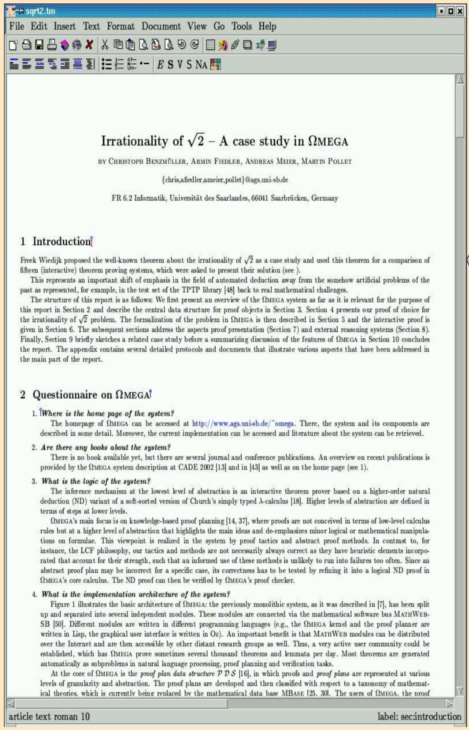
Mathemat. Knowledge Management

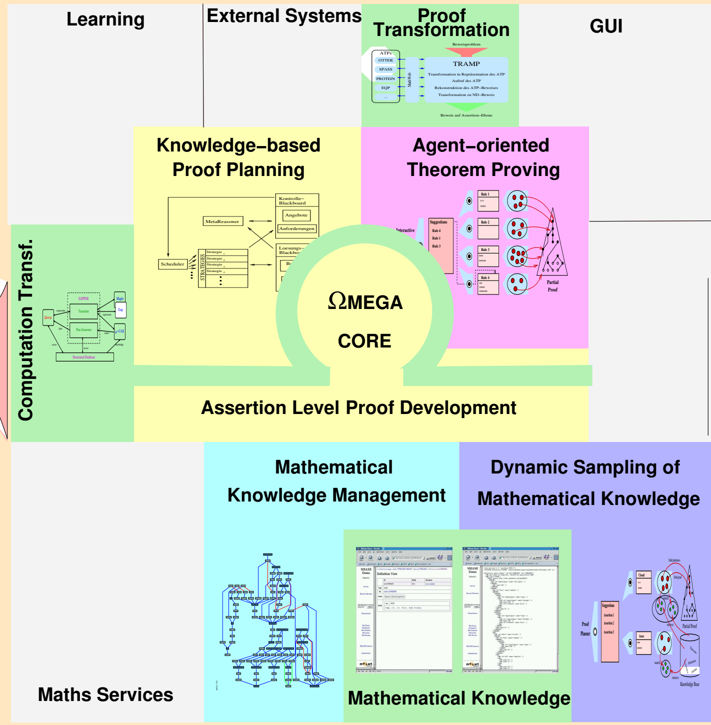
1. Types of knowledge
2. Distributed over different physical locations
3. Evolution of mathematical knowledge
   - Management of change
     Benefit from experience with MAYA
   - Versioning

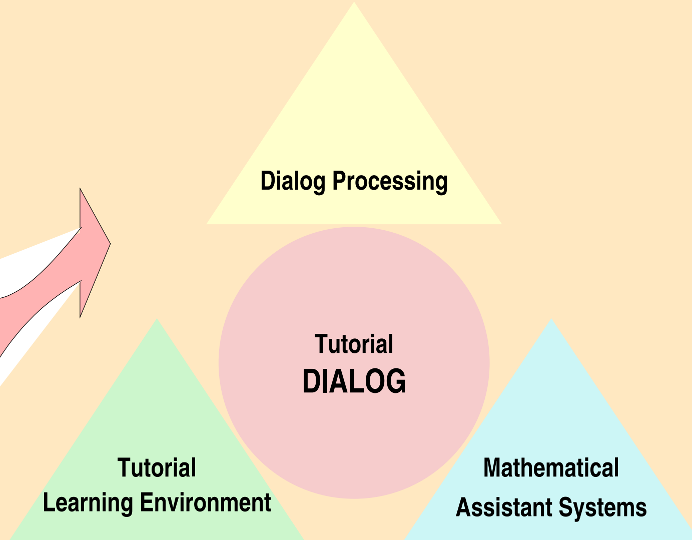




Mathematical Activities







Assistance in Preparing Mathematical Documents

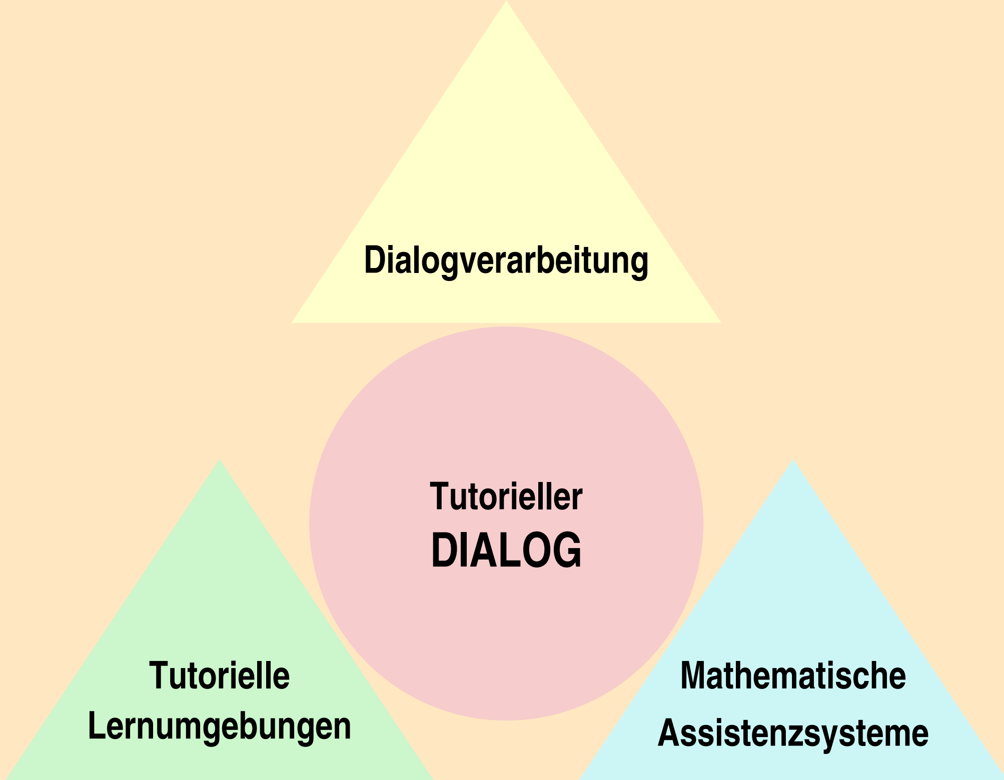
Technical Support for Tutoring Tools

Source: Autexier & Benzmüller & Pollet

DIALOG

Natural Language-based Interaction with a Mathematics Assistance System

The DIALOG Project

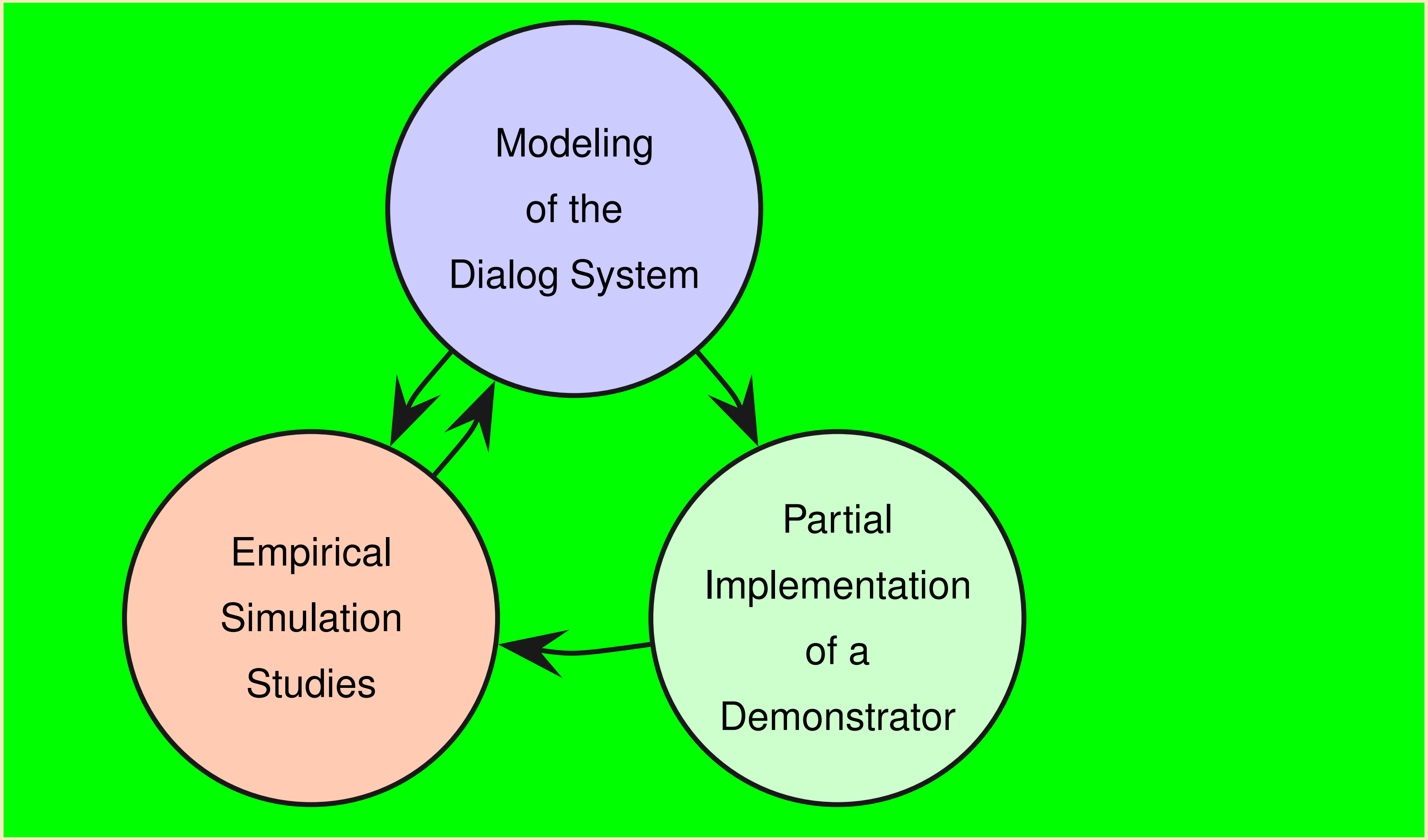


- Joint project (between Coli and CS) as part of the SFB378 on *Resource-adaptive cognitive processes*
- Selected mathematical domain: naive set theory

Team

- Directly Involved Researchers: Manfred Pinkal (Coli), Jörg Siekmann (CS), Christoph Benzmüller (CS), Ivana Kruijff-Korbayova (Coli), Magdalena Wolska (Coli), Quoc Bao Vo (CS), Armin Fiedler (CS), Dimitra Tsovaltzis (CS)
- Collaborators: Serge Autexier (CS), Malte Gabsdil (Coli), Helmut Horacek (CS), Alexander Koller (Coli), Erica Melis (CS)
- Hiwis and Students: Mark Buckley (CS), Hussain Syed Sajjad (CS), Marvin Schiller, Jochen Setz (CS), Michael Wirth (Coli), Sreedhar Ellisetty (CS), Andrea Schuch (Coli), Beata Biehl (Coli), Oliver Culo (Coli)

Method: Progressive Refinement



Source: Autexier & Benzmüller & Pollet

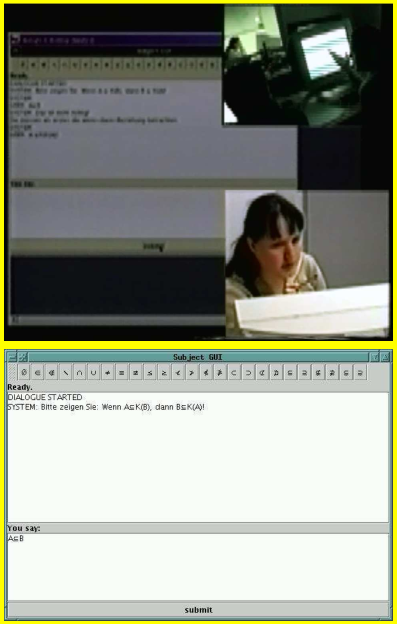
WOZ-Experiment $\longrightarrow$ Own Corpus

- collect data
  (for analysis of phenomena)
- test coarse grained model of system
- test hinting algorithm

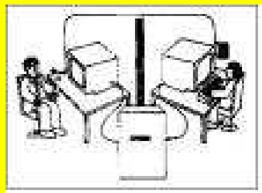
- 24 Subjects:
  - university students
  - varying background
  - varying math knowledge
- Wizard:
  - mathematician with tutoring experience
  - assisted by developers of hinting algorithm
- Experimenter

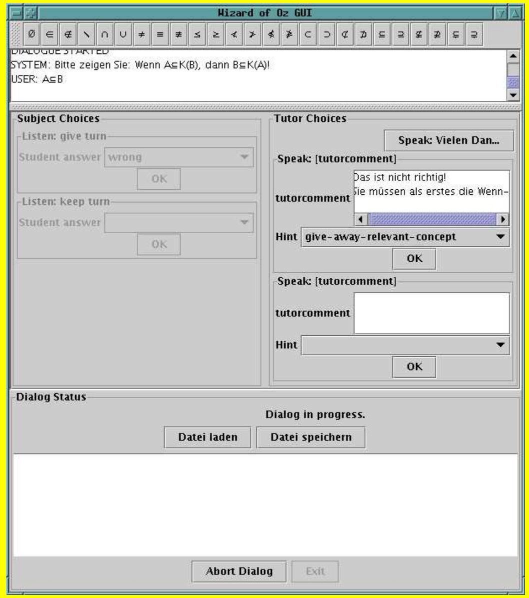
Source: Autexier & Benzmüller & Pollet

WOZ-Experiment ⟶ Own Corpus



The subject window





The wizard window

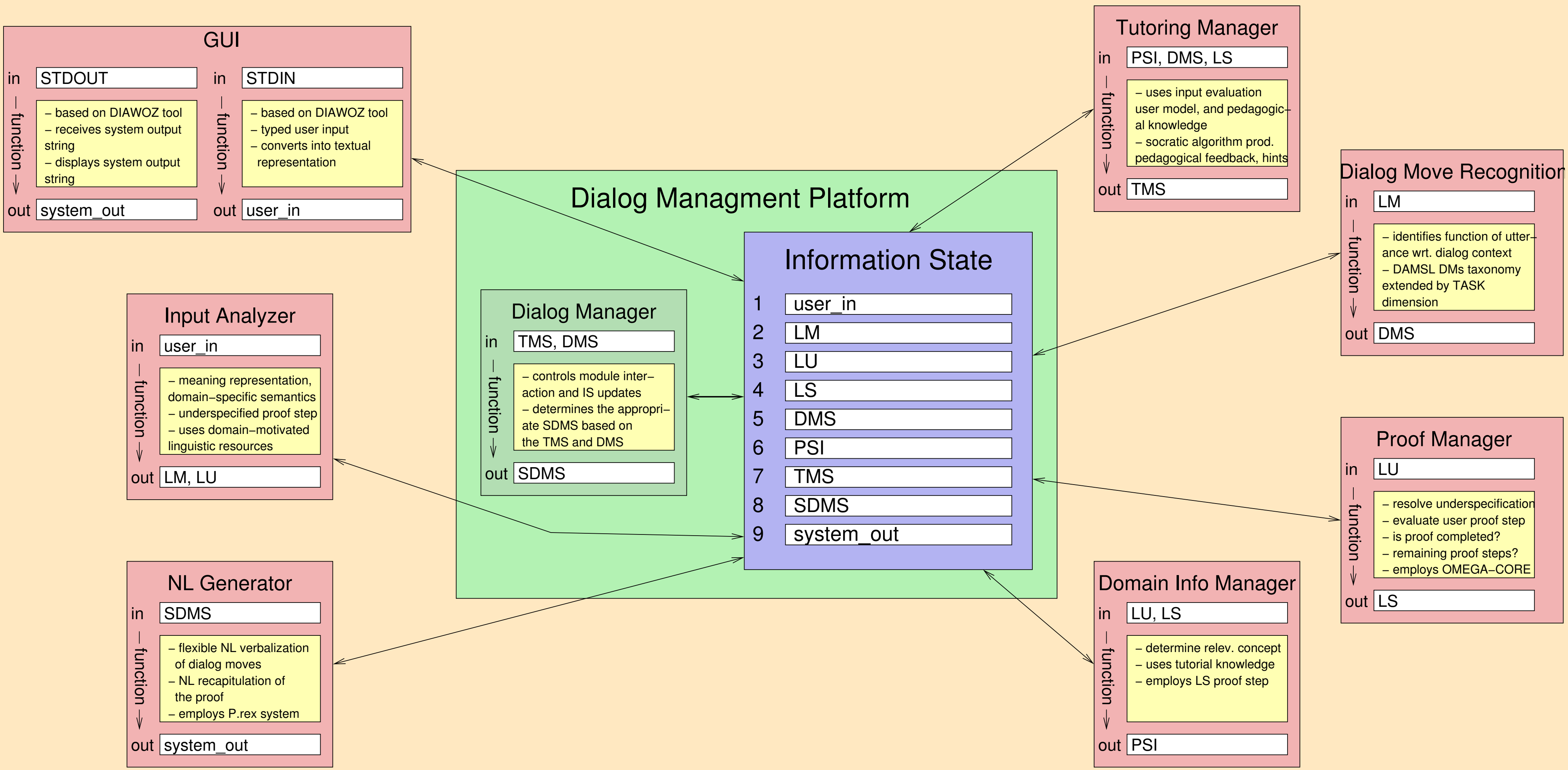
Corpus Example

Example: didactic, vp16, dryrun

T1: Bitte zeigen Sie : $K((A \cup B) \cap (C \cup D)) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$!</s>
S1: (correct) nach deMorgan-Regel-2 ist $K((A \cup B) \cap (C \cup D)) = (K(A \cup B) \cup K(C \cup D))$</s>
T2: Das ist richtig !</s>
S2: (correct) $K(A \cup B)$ ist laut DeMorgan-1 $K(A) \cap K(B)$</s>
T3: Das stimmt auch .</s>
S3: (correct) und $K(C \cup D)$ ist ebenfalls laut DeMorgan-1 $K(C) \cap K(D)$</s>
T4: Auch das stimmt .</s>
S4: (correct) also folgt letztendlich : $K((A \cup B) \cap (C \cup D)) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$.</s>
T5: Das stimmt genau .</s> Ich wiederhole noch einmal : Auf die linke Seite der Gleichung kann ich zuerst die zweite und danach die erste de-Morgan-Regel anwenden , so daß sich folgende Argumentationskette ergibt : $K((A \cup B) \cap (C \cup D)) = K(A \cup B) \cup K(C \cup D) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$.</s>

Source: Autexier & Benzmüller & Pollet

Demonstrator Architecture



Source: Autexier & Benzmüller & Pollet

Example Utterance

T1: Bitte zeigen Sie : K ((A ∪ B) ∩ (C ∪ D)) = (K (A) ∩ K (B)) ∪ (K (C) ∩ K (D)) !</s>

S1: (correct) nach deMorgan-Regel-2 ist K ((A ∪ B) ∩ (C ∪ D)) = (K (A ∪ B) ∪ K (C ∪ D))</s>

T2: Das ist richtig !</s>

Project Progress: First Phase

- Experiment & WOZ Tool & Corpus
- NL Analysis
- Tutoring Aspects
- Dialog Management
- Proof Manager
- Proof Step Evaluation
- Modeling of System & Demonstrator

Overview, Papers, Corpus, etc.: see

http://www.ags.uni-sb.de/~chris/dialog/

Proof Manager: Tasks

- Resolution of
  - Ambiguities
  - Underspecification
- Proof Step Evaluation
  - Soundness: Can the proof step be verified by a formal inference system?
  - Granularity: Is the granularity (i.e., 'logical size' or 'argumentative complexity') of the proof step acceptable?
  - Relevance: Is the proof step needed or useful in achieving the goal?

Source: Autexier & Benzmüller & Pollet

Ambiguities and Underspecification

Example	*Where does ambiguity arise?*	*Ambiguity resolution means*
(1) $x \in B$ und somit $x \subseteq K(B)$ und $x \subseteq K(A)$ wegen Vorraussetzung (2) A enthaelt B	linguistic meaning level; attachment, coordination	linguistic means; type checking in (2)
(3) P((A∪C)∩(B∪C))=PC ∪(A∩B) (4) K((A∪C)∩(B∪C))=KC ∪(A∩B)	linguistic meaning level; informal character of discourse	type checking for (3); domain reasoning for (4)
(5) T1: Bitte zeigen Sie: $K((A \cup B) \cap (C \cup D)) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$! S1: nach deMorgan-Regel-2 ist $K((A \cup B) \cap (C \cup D)) = (K(A \cup B) \cup K(C \cup D))$	underspecified proof step	domain reasoning

Example: Proof Step Evaluation

Assertions already introduced

(A1) $A \wedge B$.

(A2) $A \Rightarrow C$.

(A3) $C \Rightarrow D$.

(A4) $F \Rightarrow B$.

(G) $D \vee E$.

Alternative proof step directives.

(a) Aus den Annahmen folgt D.

(b) B gilt.

(c) Es genügt D zu zeigen.

(d) Wir zeigen E.

Proof Step Evaluation

Criterion	*Task (first approach)*	*Requirements for theorem prover*
Soundness	$E \vdash^{?}_{C} D \vee E$	'Yes' or 'No' answer; any theorem prover resp. calculus C
Granularity	proof-steps($E \vdash^{?}_{C} D \vee E$)	adequate abstract-level theorem prover resp. calculus C; measure 'shortest' proof; take tutorial constraints into account; proof planning or assertion level reasoning?
Relevance	$\begin{array}{l} A \wedge B \\ A \Rightarrow C \\ C \Rightarrow D \\ F \Rightarrow B \end{array} \vdash^{?}_{C} E$	recognize detours; compare with other 'shorter' proofs; take tutorial constraints into account; forward case more challenging

Source: Autexier & Benzmüller & Pollet

What are the right provers?

???

DIALOG project is (from a cognitive science perspective) a

$\rightarrow$ very fascinating, novel application domain for deduction systems.