ΩMEGA

A Mathematics Assistance System

and

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

Christoph Benzmüller and Martin Pollet

Saarland University, Saarbrücken, Germany

October 20th 2004

Cambridge, England

Overview

ΩMEGA

- Mathematics Assistance Environments
- The ΩMEGA Project
  - Mathematics Assistant In-the-small

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

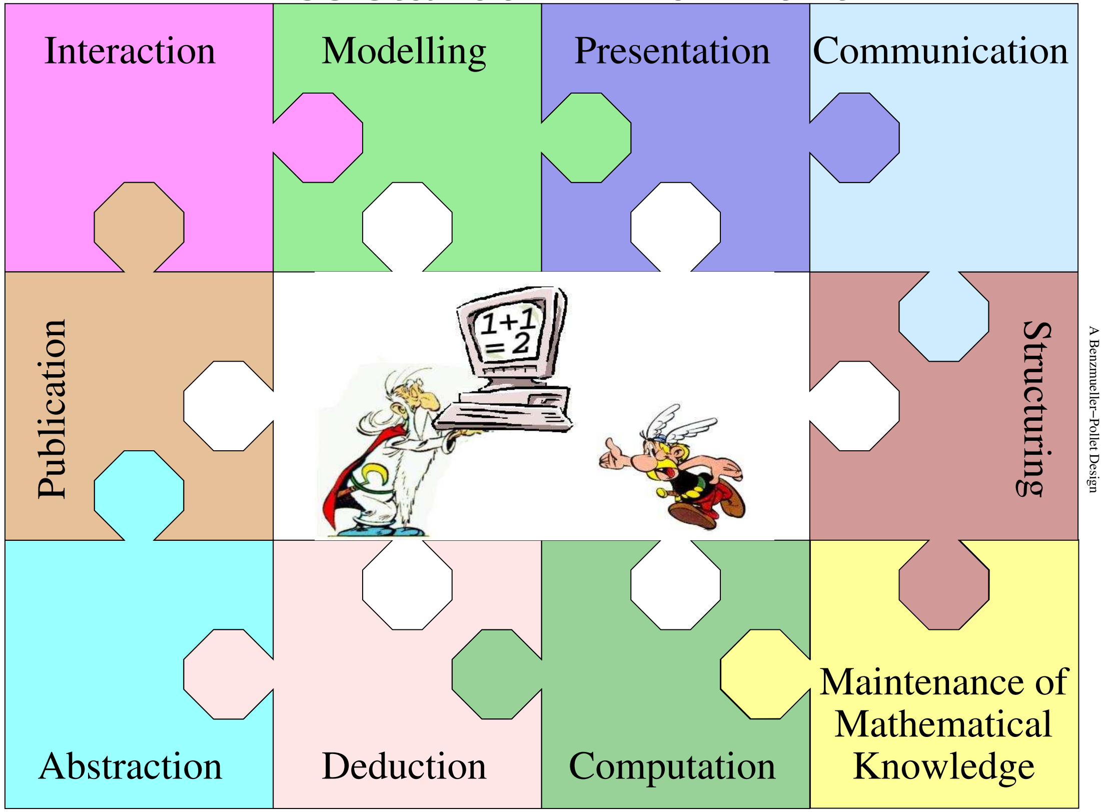
    – novel research directions –
- DIALOG: Natural Language-based Interaction with a Mathematics Assistance System

UNIVERSITÄT DES SAARLANDES

Mathematics Assistance Systems

ΩMEGA

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



Join of ressources necessary

Applications

Mathematics research
Mathematics education
Formal methods

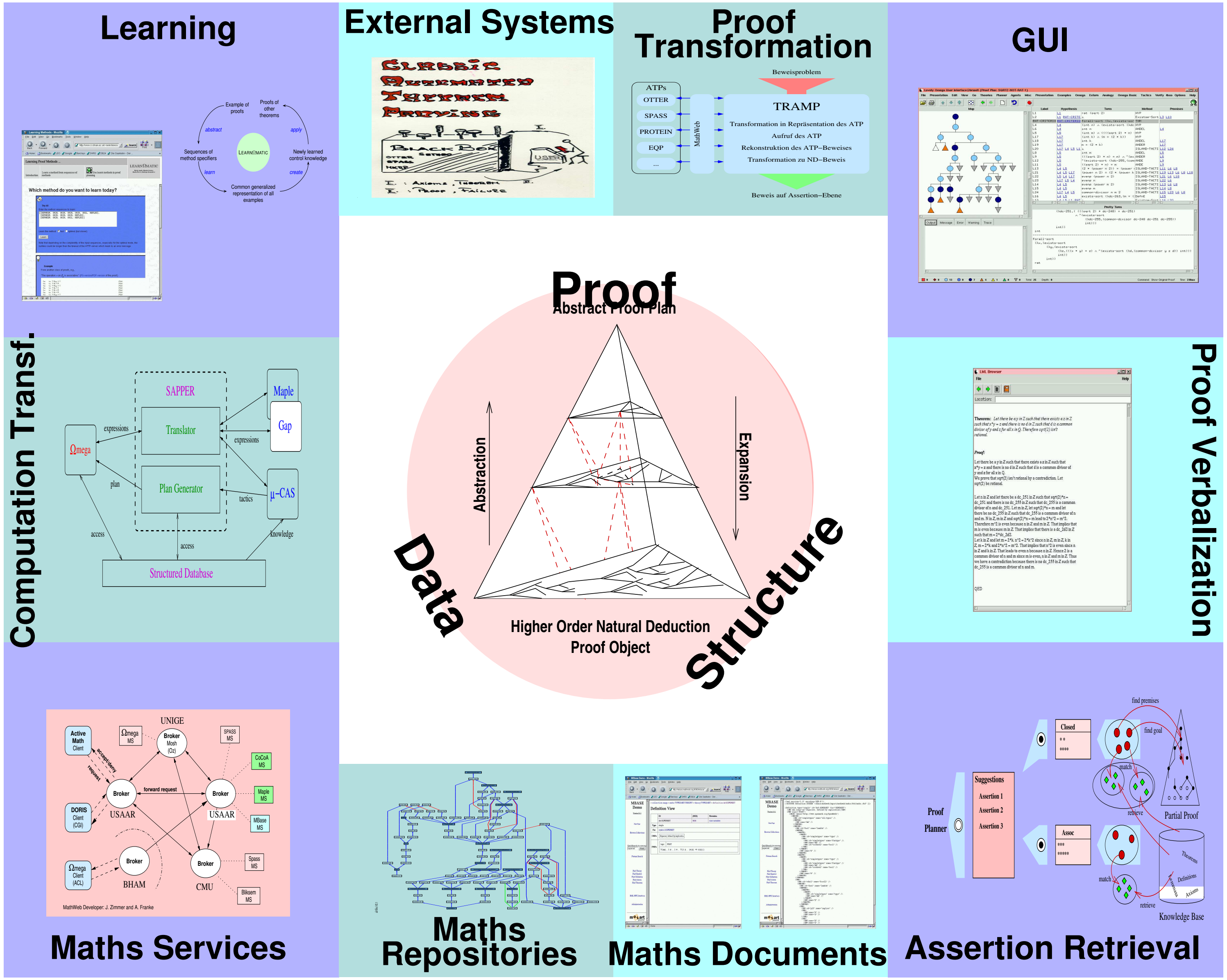
System level: Coq, NuPrl, Isabelle/HOL, PVS, Theorema, ΩMEGA, CLam, ...

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

Mathematics Assistance System In-the-small

Research directions in the ΩMEGA project since the early 90s

The ΩMEGA Instance



MBASE & OMDOC

ΩMEGA

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

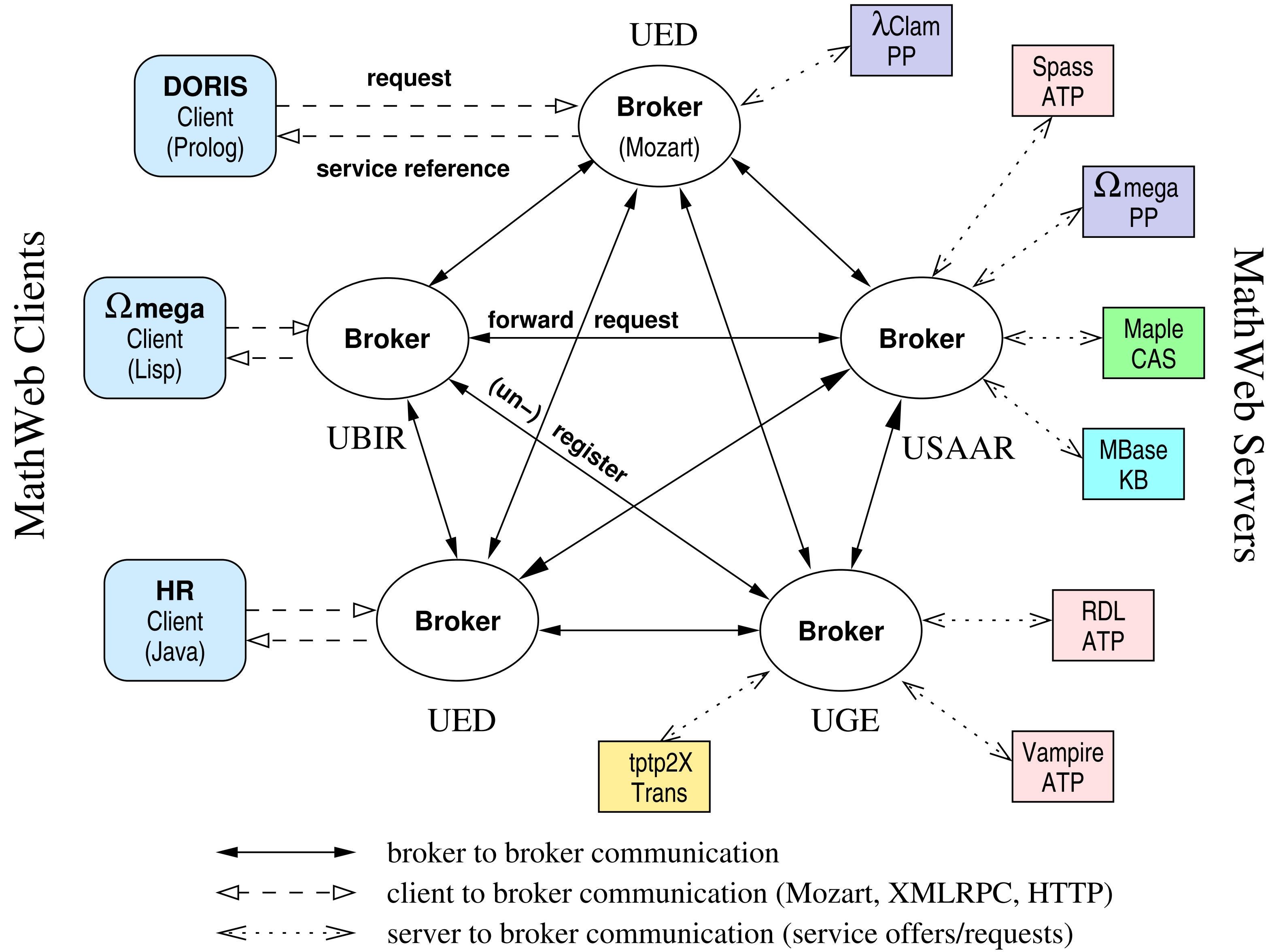
– Version control: concurrent, joint development of mathematical knowledge

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

UNIVERSITÄT DES SAARLANDES

External Specialist Reasoners

ΩMEGA



UNIVERSITÄT DES SAARLANDES

External Specialist Reasoners

ΩMEGA

Usually required in OMEGA:

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

available for:

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

External Specialist Reasoners

ΩMEGA

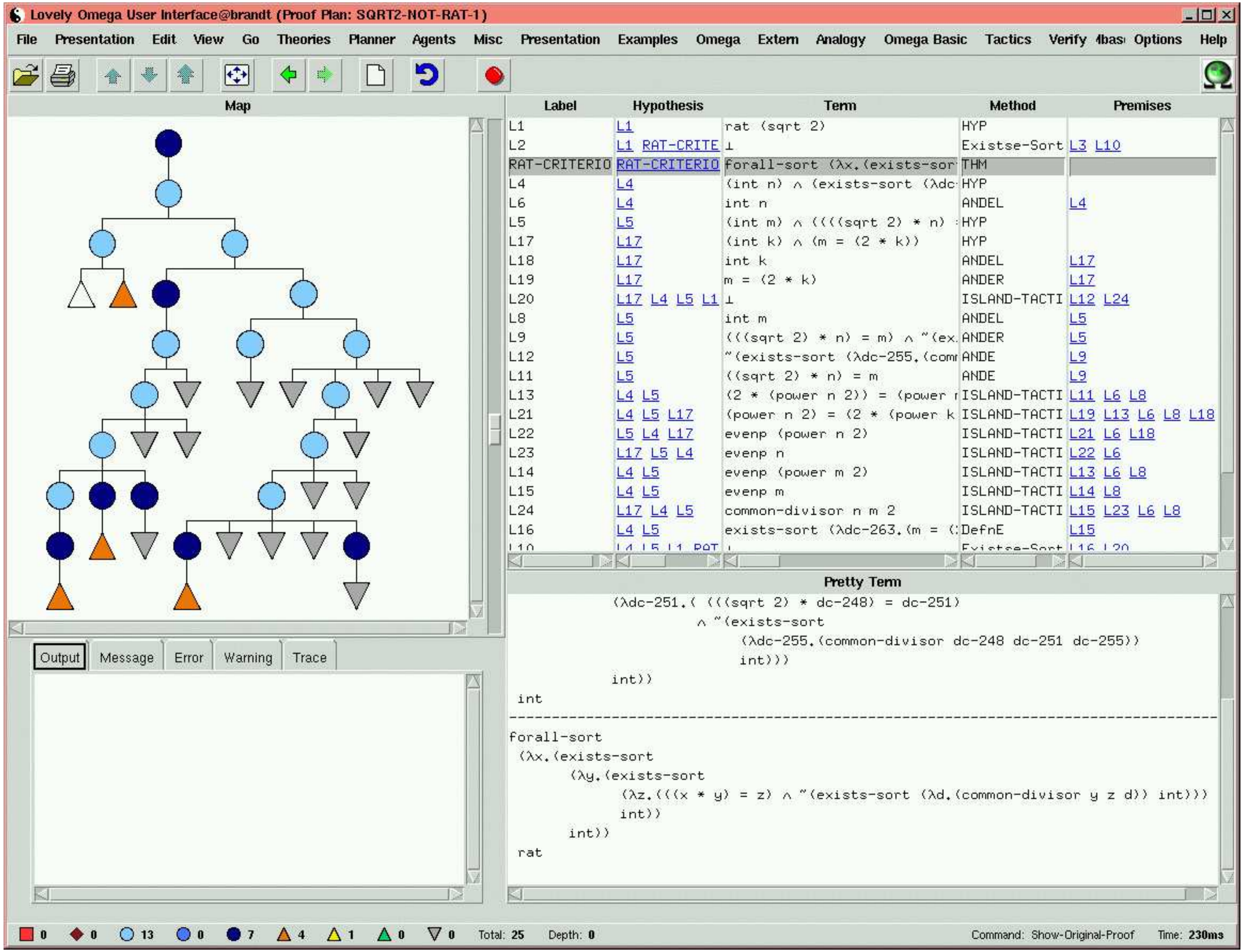
Discussion

+ 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

UNIVERSITÄT DES SAARLANDES

User Interface

ΩMEGA



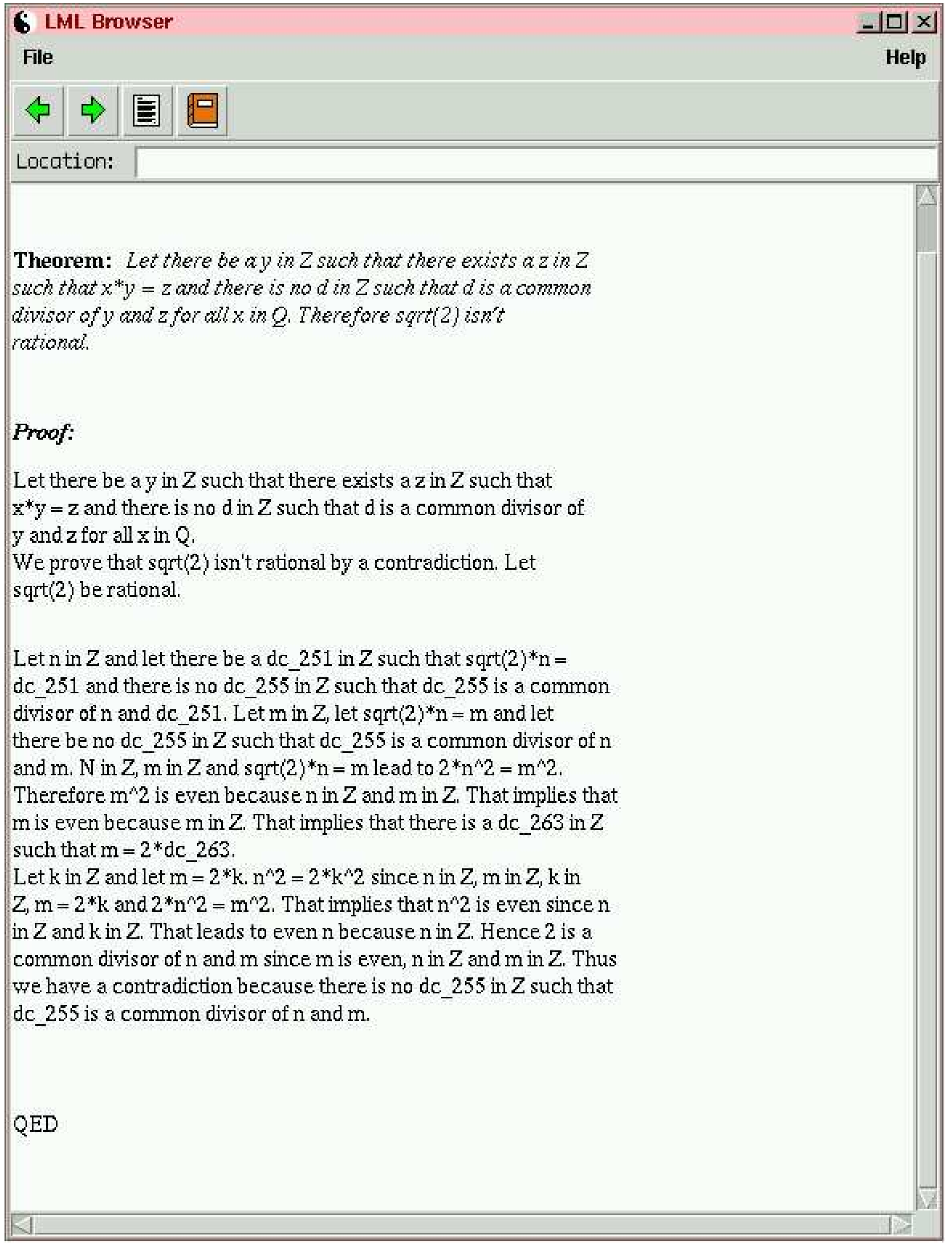
UNIVERSITÄT DES SAARLANDES

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

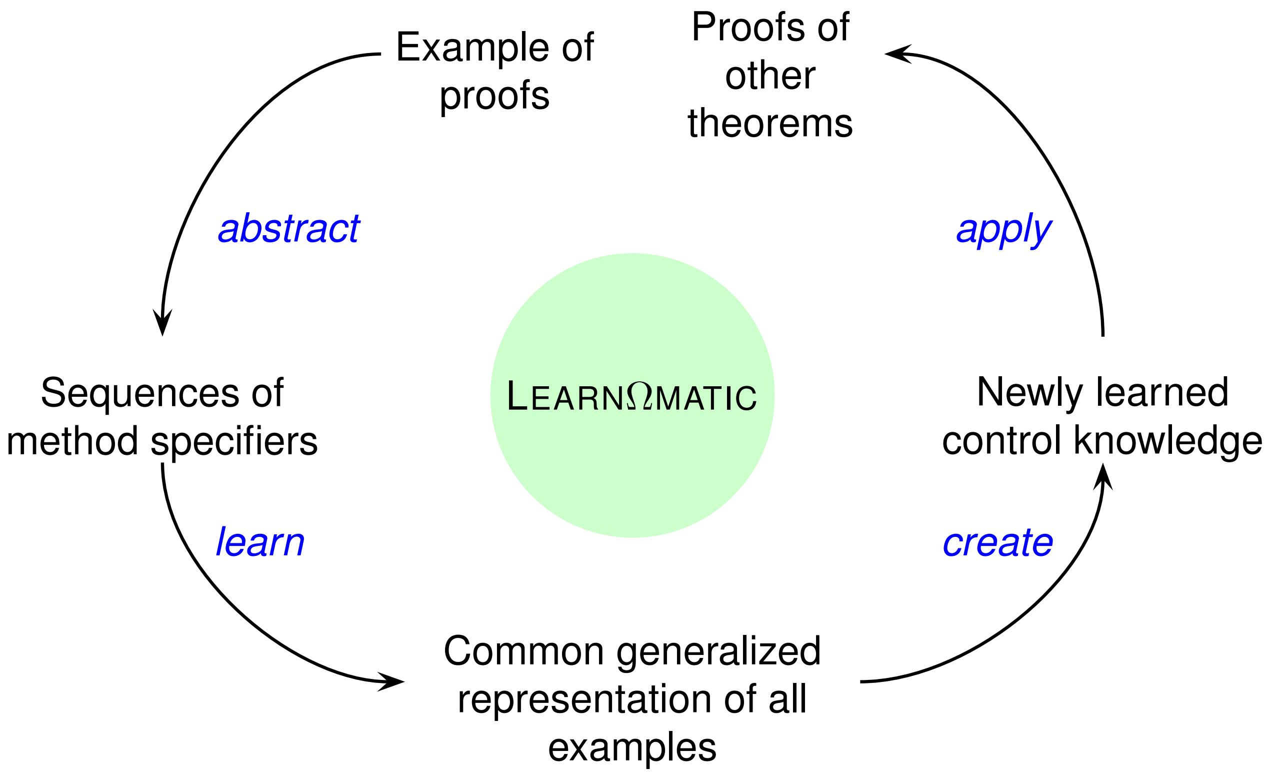
Proof Verbalization

Discussion

+ Flexible, adaptable, non-template based proof verbalization

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

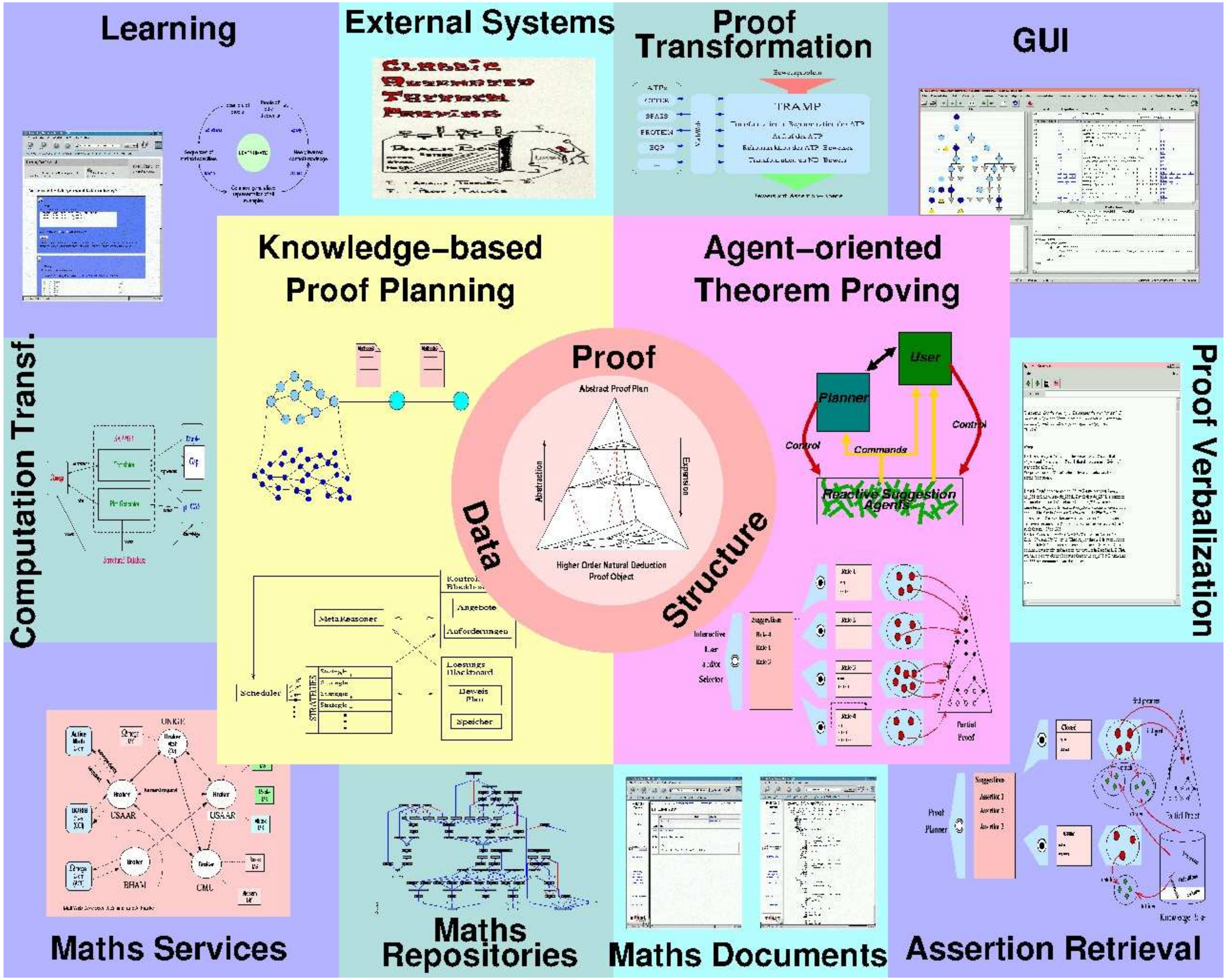
Hybrid system consisting of learn engine and deduction system.



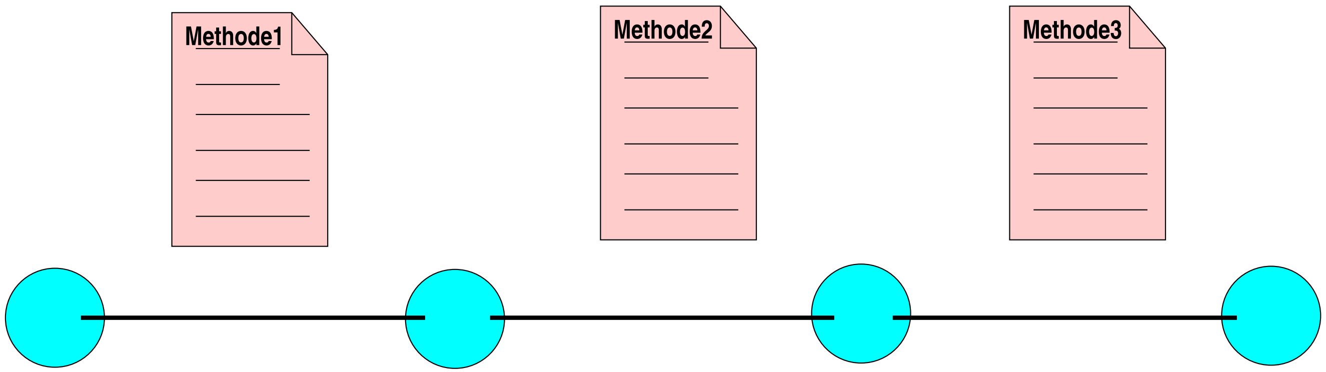
+ More theorems provable, proof search more directed, and shorter proofs

– What other information can be considered for learning?

The Complete Picture



Proof Planning



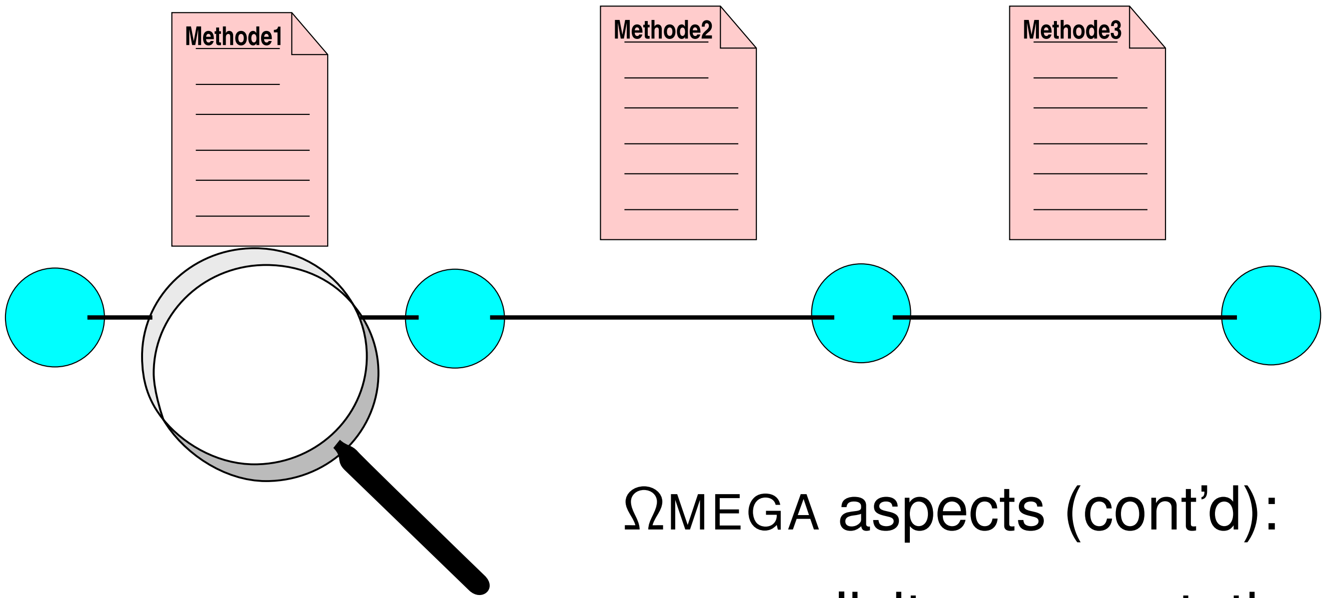
ΩMEGA born in early 90s; inspired by [Bundy88]

- 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

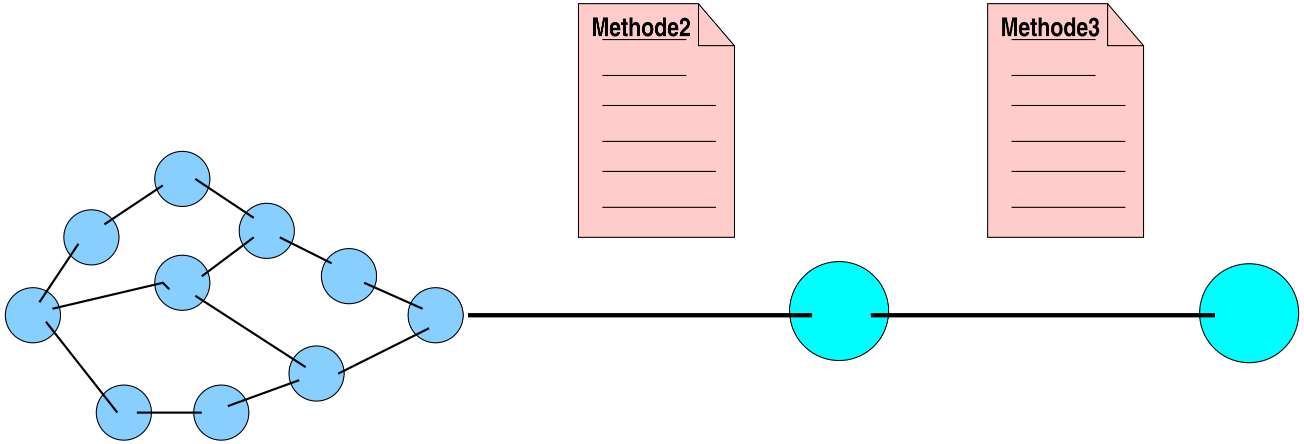
Proof Planning



ΩMEGA aspects (cont'd):

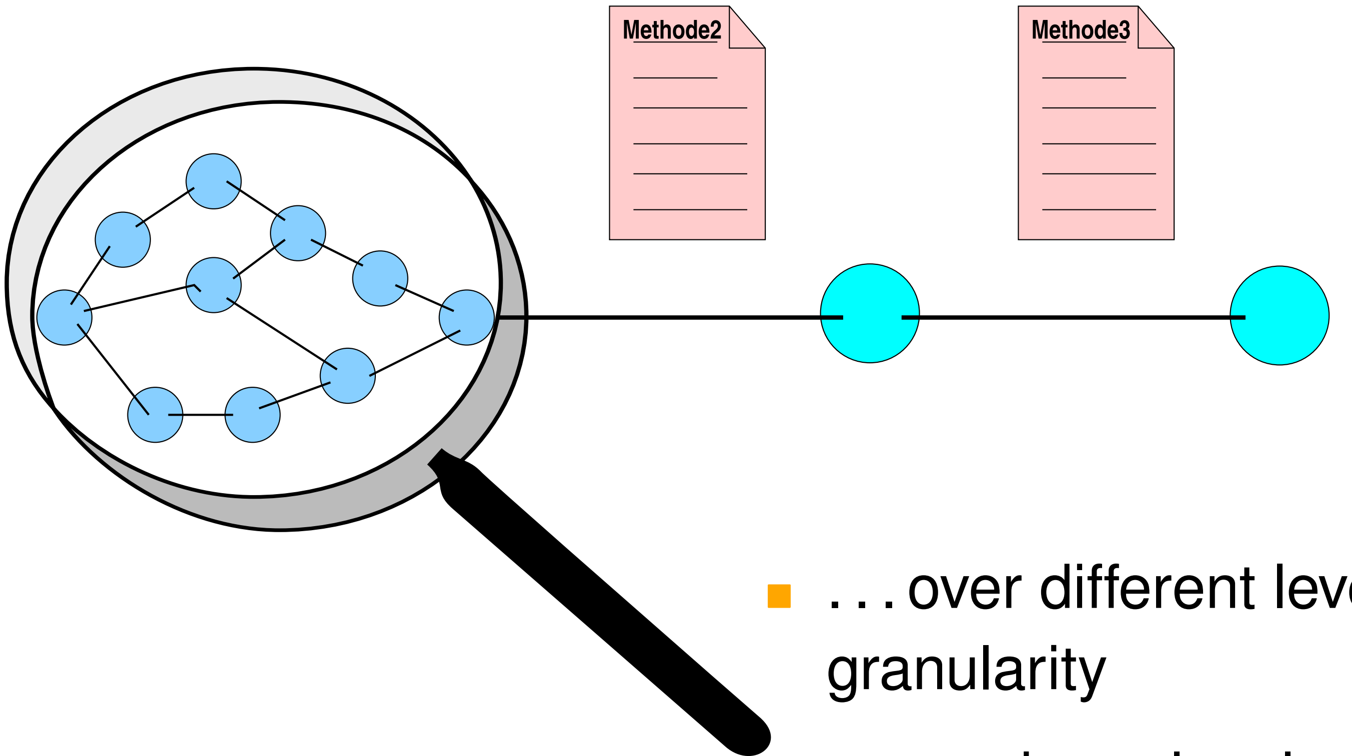
- explicit representation of proof plans
- under-specification of pre-conditions: potentially non-sound proof plans
- soundness guaranteed via . . .

Proof Planning



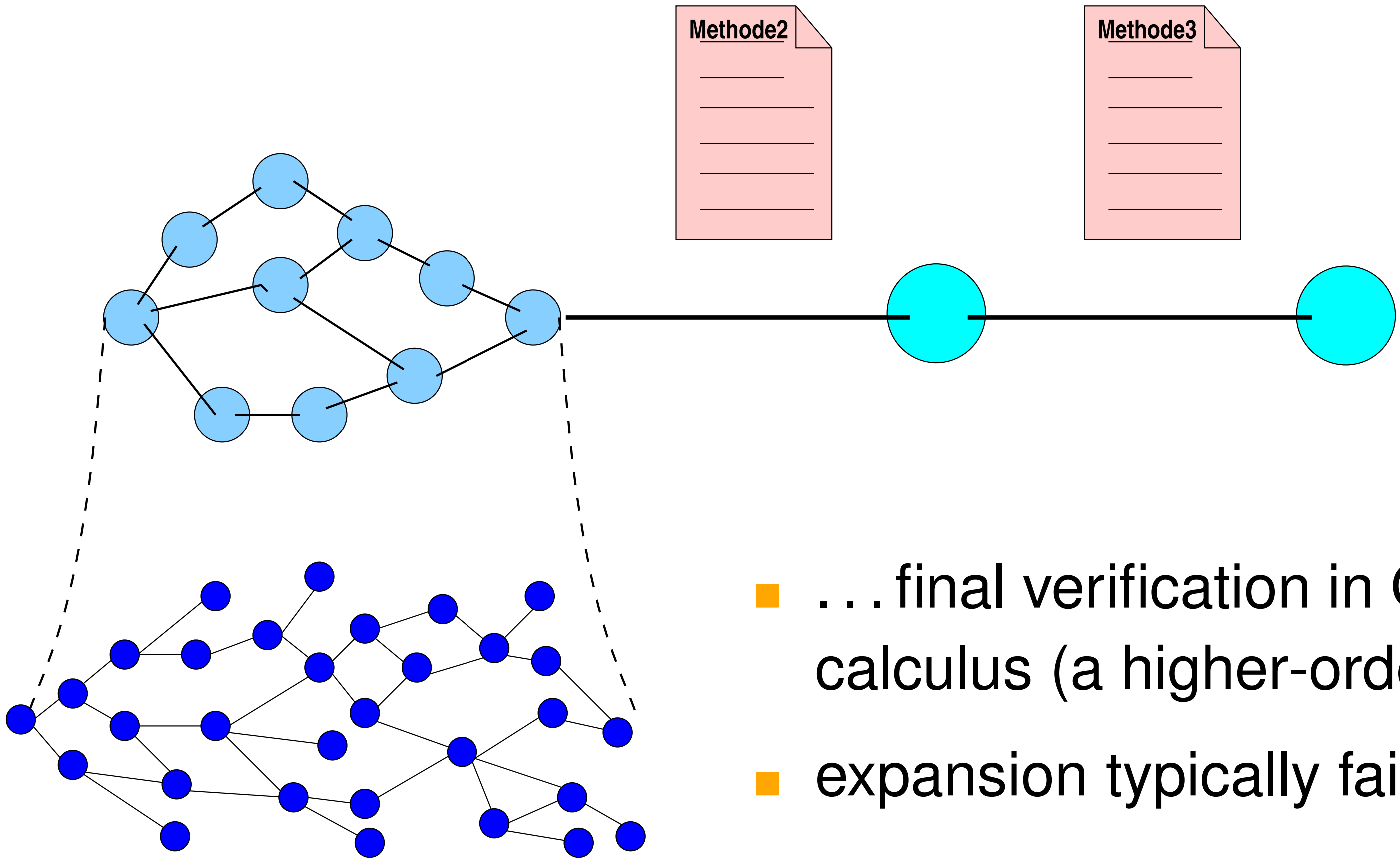
- . . . proof (plan) expansion over . . .

Proof Planning



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

Proof Planning



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

Proof Planning

ΩMEGA

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

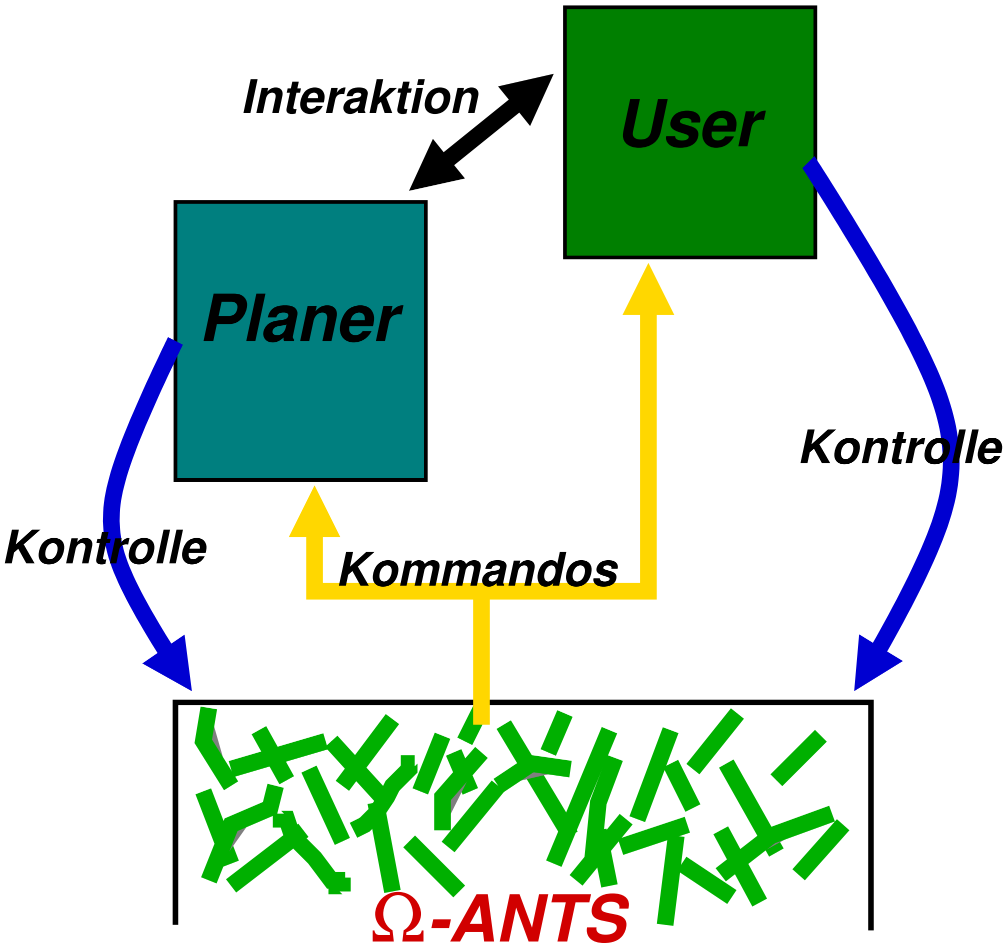
Discussion

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

..

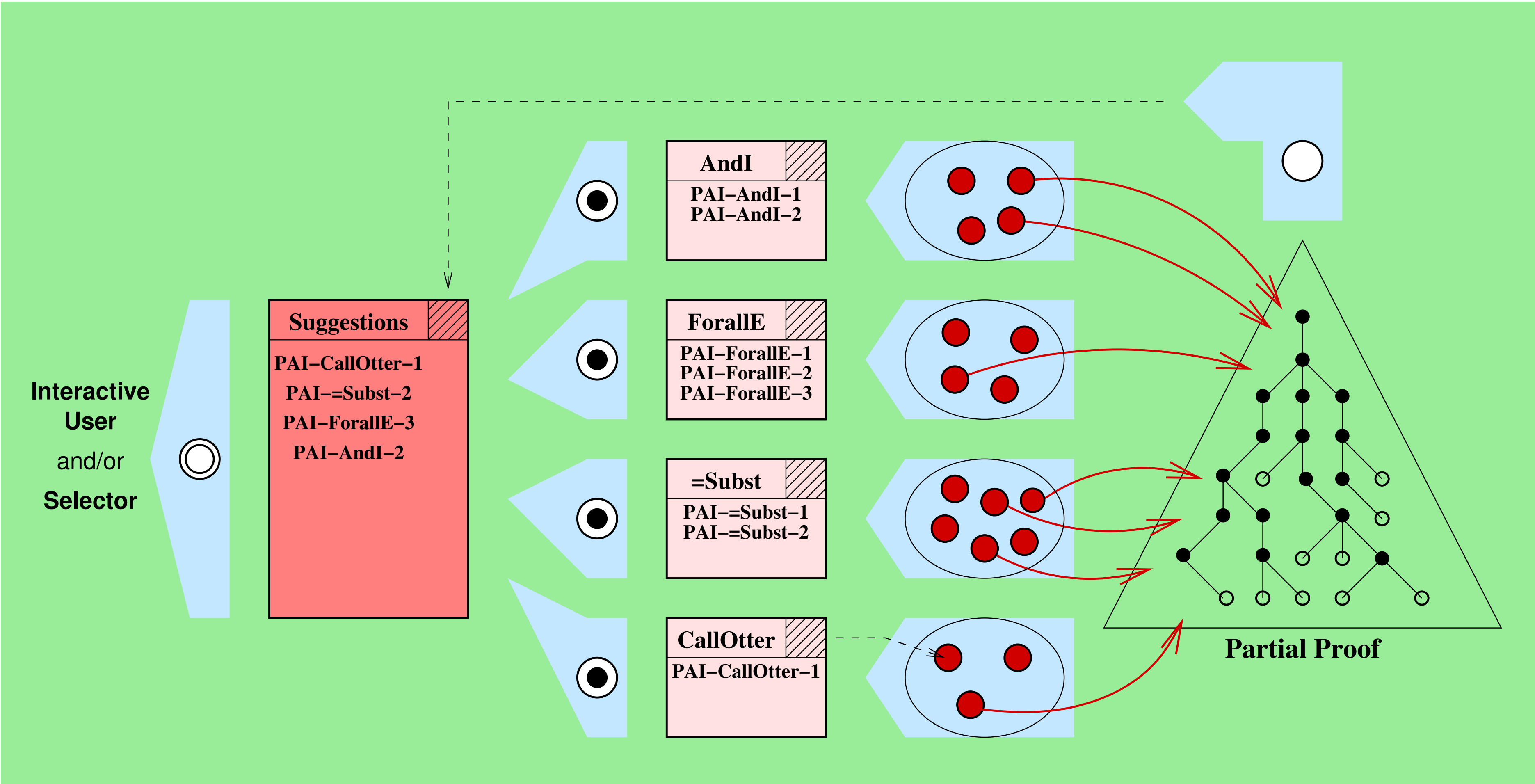
UNIVERSITÄT DES SAARLANDES

Agent-based Theorem Proving



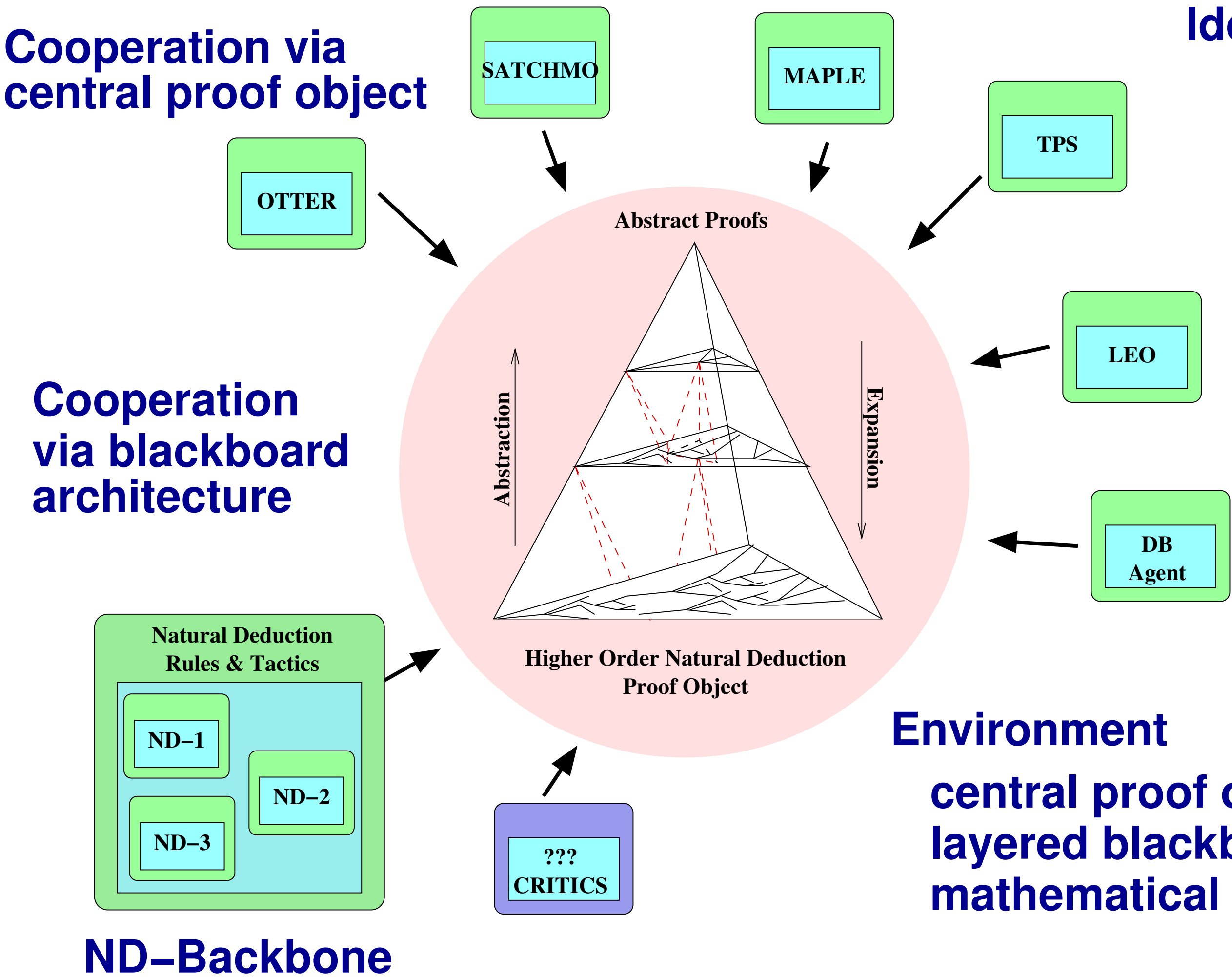
- pro-active support versus passive support
- concurrent versus sequential

Agent-based Theorem Proving



Agent-based Theorem Proving

ΩMEGA



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

Agent-based Theorem Proving

ΩMEGA

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

Novel Research Directions

Mathematics Assistance System In-the-Large

UNIVERSITÄT DES SAARLANDES

Current & Future Developments

ΩMEGA

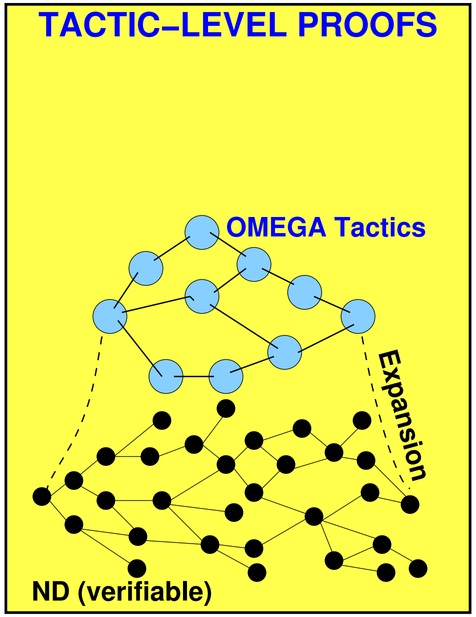
Theme: **Towards a smoother integration into spectrum of typical mathematical activities**

- 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

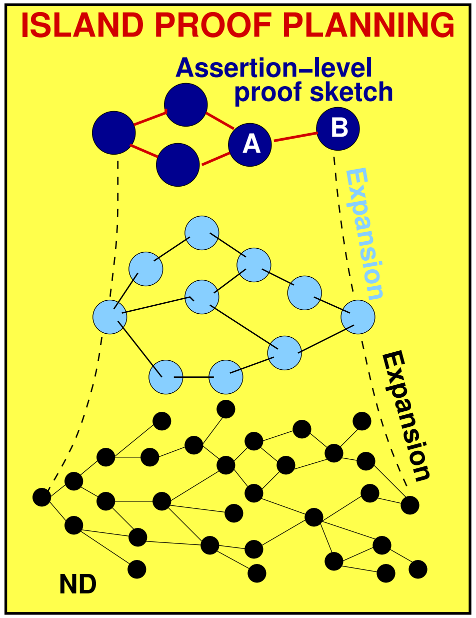
like PVS, ISABELLE, COQ, etc.

(and OMEGA) can construct a proof –

but not at an adequate level

Proof Planning & The CoRe Calculus

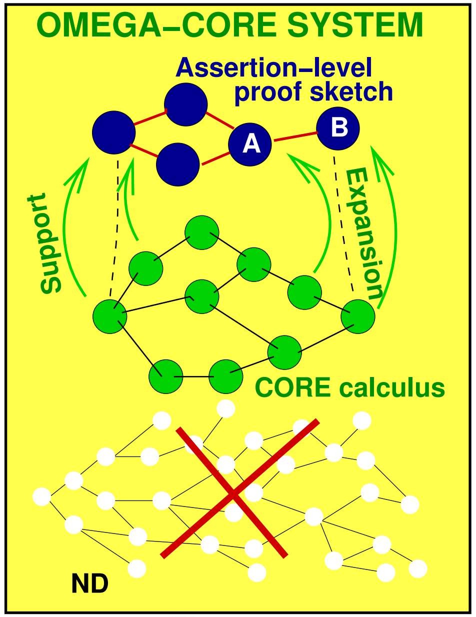
ΩMEGA



- 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

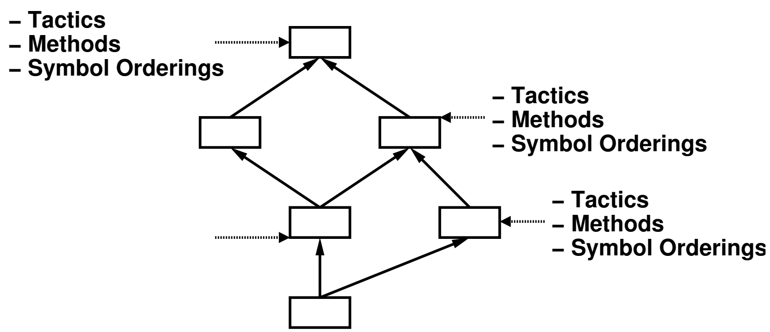


- CORE is more natural/abstract-level base calculus for proof assistants
- subsumes old ND base and (parts of) the old tactic layer
- provides more direct, constructive support for upper layer(s)

UNIVERSITÄT DES SAARLANDES

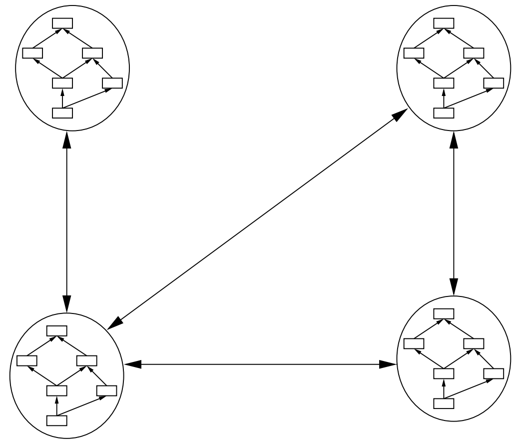
Mathematical Knowledge Management

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



Mathematical Knowledge Management

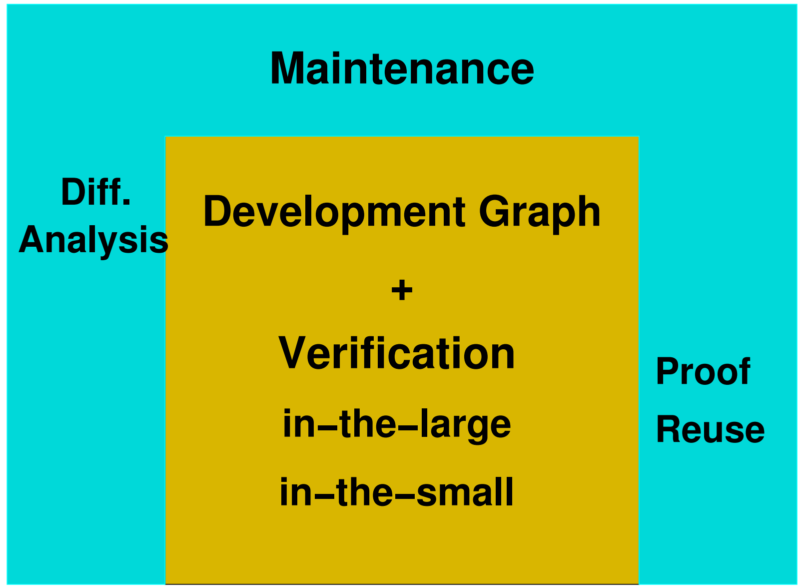
1. Types of knowledge
2. Distributed over different physical locations
   - Origin tracking, remote access, . . .

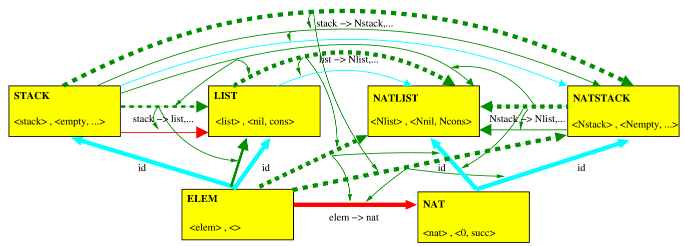


Mathematical Knowledge Management

ΩMEGA

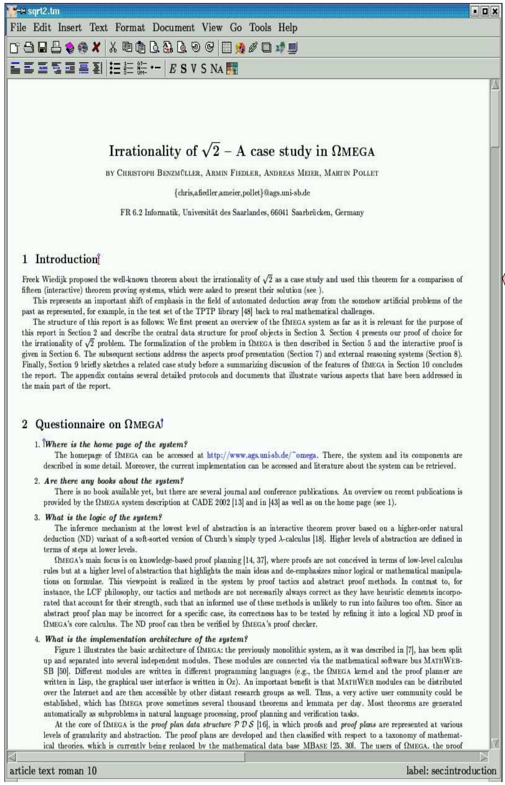
1. Types of knowledge
2. Distributed over different physical locations
3. Evolution of mathematical knowledge
   - Management of change
     Benefit from experience with MAYA
   - Versioning

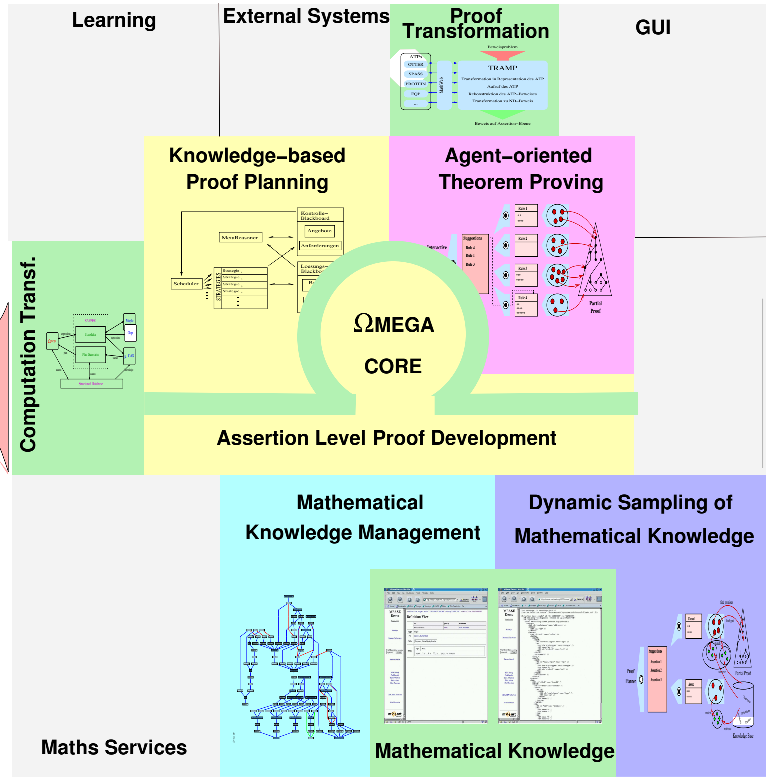


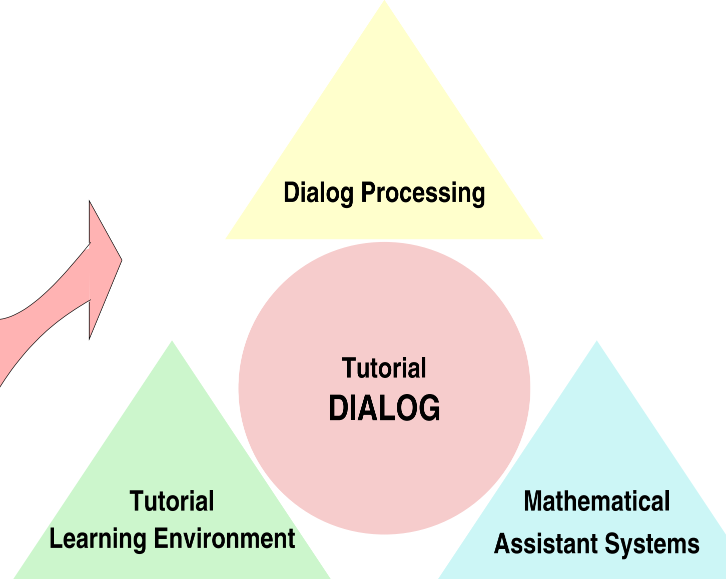


Integration into Mathematical Activities

ΩMEGA







Assistance in Preparing Mathematical Documents

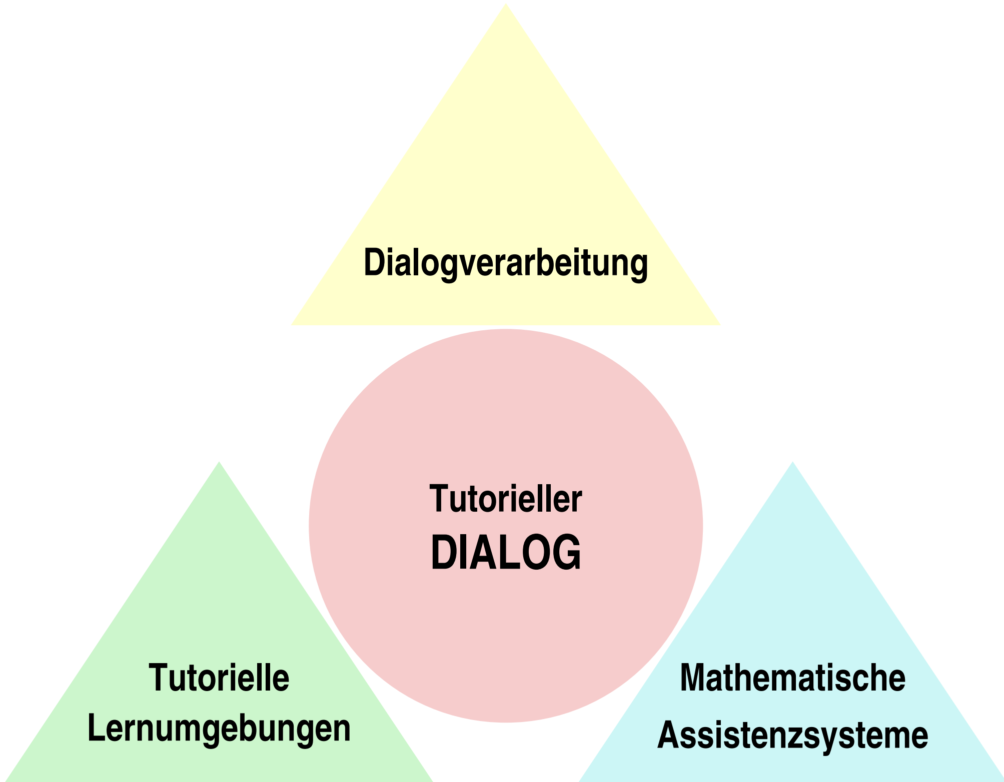
Technical Support for Tutoring Tools

UNIVERSITÄT DES SAARLANDES

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

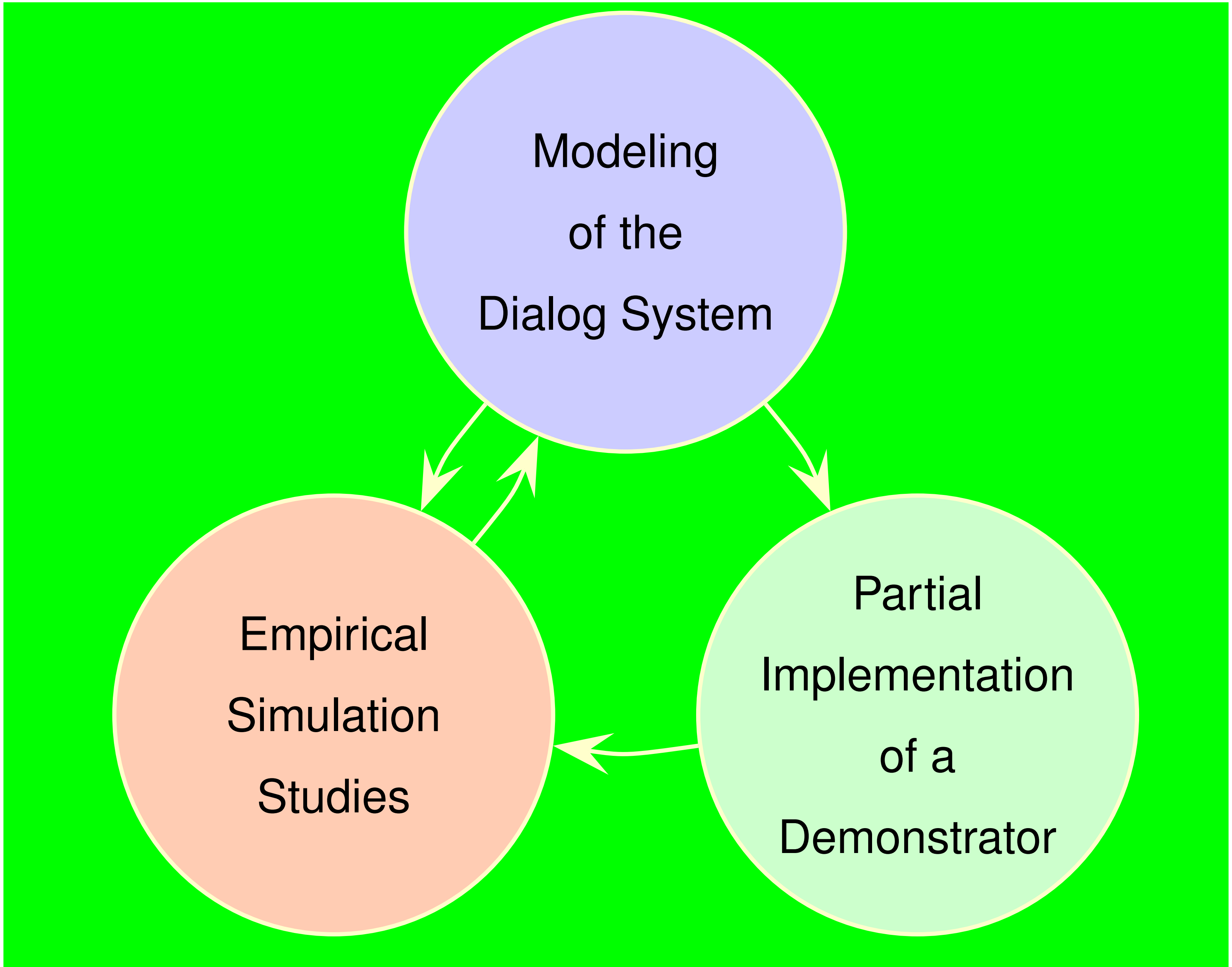
Team

ΩMEGA

- 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)

UNIVERSITÄT DES SAARLANDES

Method: Progressive Refinement



WOZ-Experiment ⟶ Own Corpus

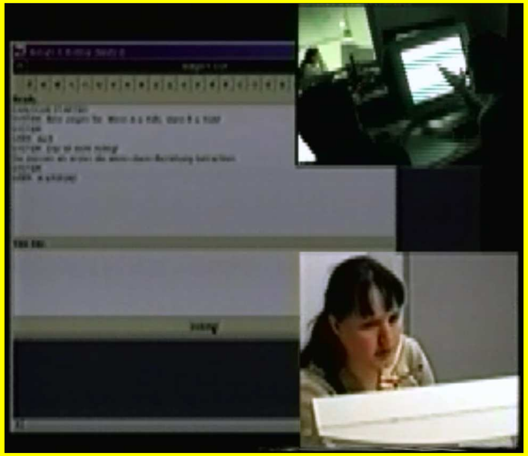
ΩMEGA

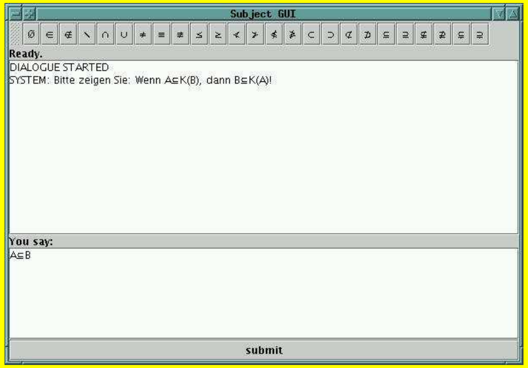
- collect data (for analysis of phenomena)
- test coarse grained model of system
- test hinting algorithm

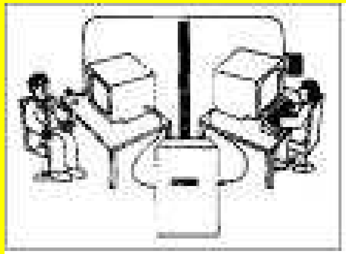
- 24 Subjects:
  - university students
  - varying background
  - varying math knowledge
- Wizard:
  - mathematician with tutoring experience
  - assisted by developers of hinting algorithm
- Experimenter

WOZ-Experiment ⟶ Own Corpus

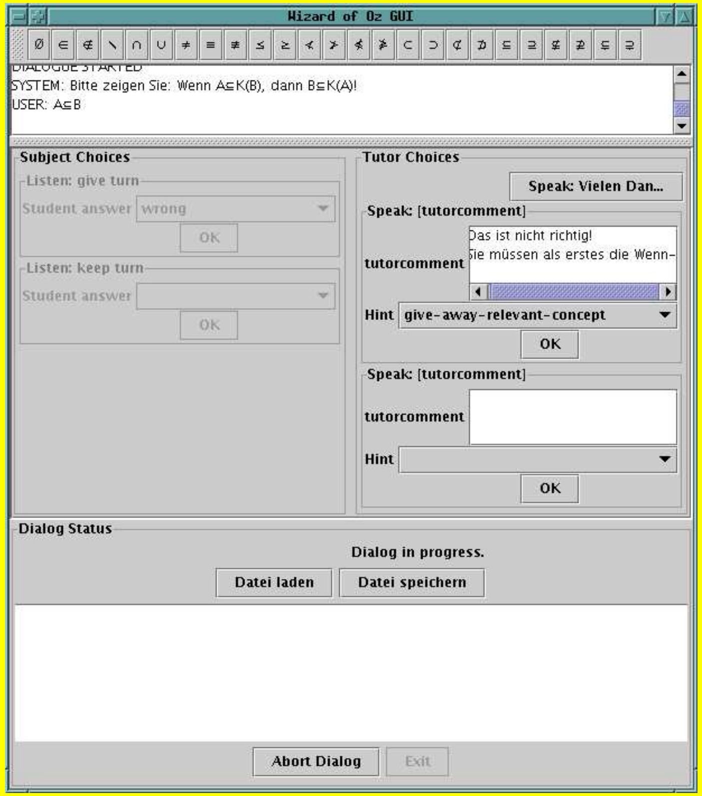
ΩMEGA







The subject window



The wizard window

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

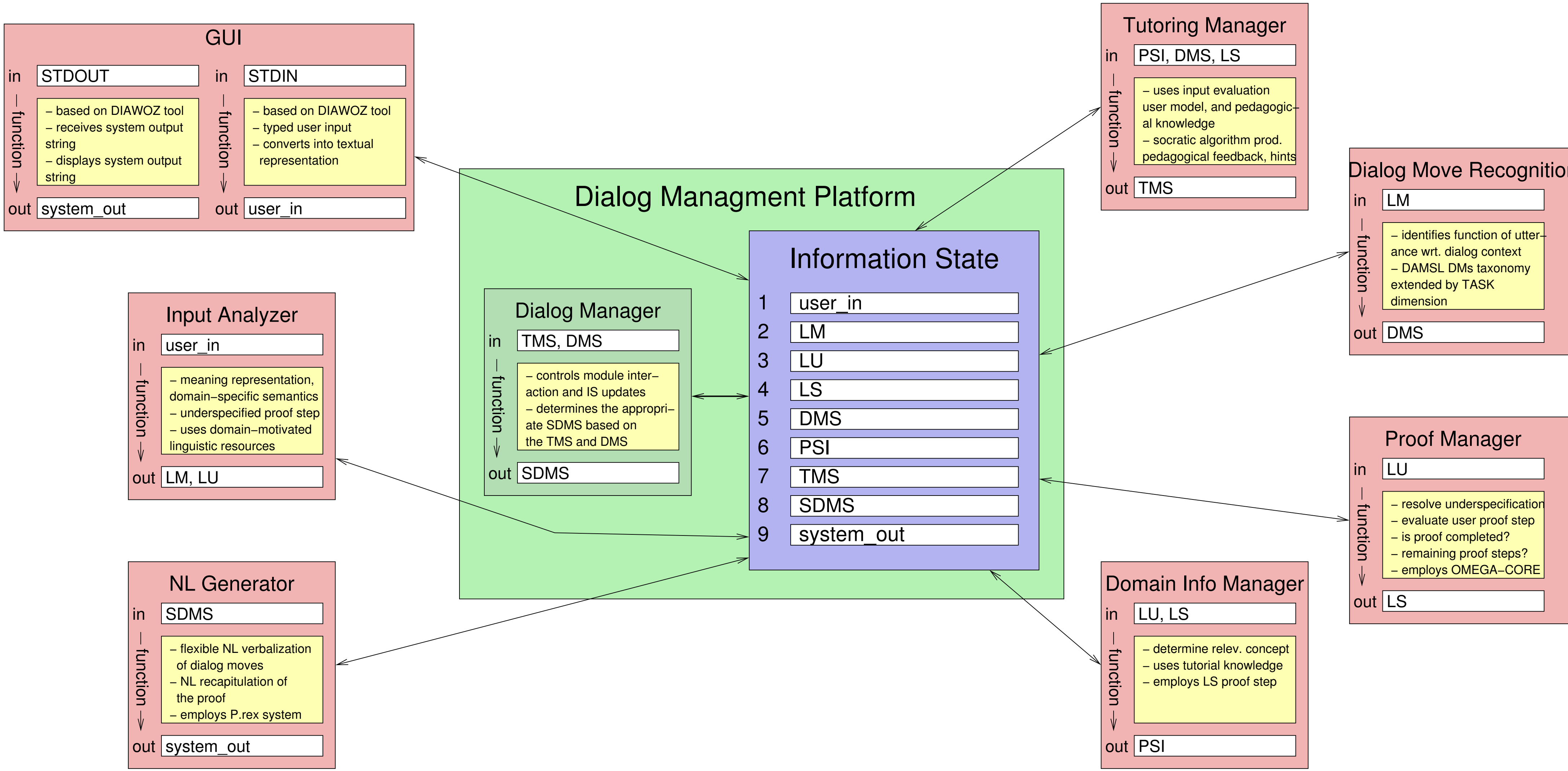
UNIVERSITÄT DES SAARLANDES

Corpus Example

Example: didactic, vp16, dryrun

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>
S2: (`correct`) K (A ∪ B) ist laut DeMorgan-1 K (A) ∩ K (B)</s>
T3: Das stimmt auch .</s>
S3: (`correct`) und K (C ∪ D) ist ebenfalls laut DeMorgan-1 K (C) ∩ K (D)</s>
T4: Auch das stimmt .</s>
S4: (`correct`) also folgt letztendlich : K ((A ∪ B) ∩ (C ∪ D)) = (K (A) ∩ K (B)) ∪ (K (C) ∩ 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 ∪ B) ∩ (C ∪ D)) = K (A ∪ B) ∪ K (C ∪ D) = (K (A) ∩ K (B)) ∪ (K (C) ∩ K (D)) .</s>

Demonstrator Architecture



Example Utterance

ΩMEGA

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>

Abbrev.	Meaning	Example
STDIN	standard input	*"nach deMorgan-Regel-2 ist* $K((A \cup B) \cap (C \cup D)) = (K(A \cup B) \cup K(C \cup D))$*"*
LM	linguistic meaning	s : @h1(holds ∧ <Norm>(d1 ∧ deMorgan–Regel–2)∧ <Patient>(f1 ∧ FORMULA1))
LU	proof language with underspecification	(input (label 1_1) (formula (= (complement (intersection (union a b) (union c d))) (union (complement (union a b)) (complement (union c d)))))) (type ?) (direction ?) (justifications (just (reference DeMorgan-2) (formula ?) (substitution ?) (role:from))))
LS	system-oriented proof language (+ evaluation)	((KEY 1_1) ⟶ ((Evaluation (expStepRepr (label 1_1) (formula (=(complement(intersection(union(A B) union(C D))) union(complement(union(A B)) complement(union(C D)))))) (type inference) (direction forward) (justification ((reference demorgan-2) (formula nil) (substitution ((X union(A B) Y union(C D)))) (role nil)))) (StepCat correct))) (ProofCompleted false) (completeProofs . . . see next poster . . .))
DMS	dialog move specification	{ fwd = "Assert", bwd = "Address_statement", commm = "", taskm = "", comms = "", task = "Domain_contribution" }
PSI	proof step information	{domConCat: "correct", proofCompleted: false, proofstepCompleted: true, proofStep: "", relConU: true, hypConU: true, domRelU: false, iRU: true, relCon: "" "+(char)8745", hypCon: "" "+(char)8746", domRel: "", iR: "deMorgan-Regel-2"}
TMS	tutorial move specification	{mode= "min"; task= (signalAccept; {proofStep= ""; relCon= ""; hypCon= ""; domRel= ""; iR= ""; taskSet= ""; completeProof= ""})}
SDMS	system dialog move specification	{ mode = "min"; fwd = "Assert"; bwd = "Address_statement"; task = ("signalCorrect", {proofStep= "", relCon= "", hypCon= "", domRel= "", iR= "", taskSet= "", completeProof= ""}); comms = ""; commm = ""; taskm = "" }
STD_out	textual representation of NL output	"Das ist richtig".

Project Progress: First Phase

ΩMEGA

- 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/`

UNIVERSITÄT DES SAARLANDES

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?

Ambiguities and Underspecification

Discourse:

(1) From previous observations we know that A or B.

(2) The former implies D by Lemma X.

(3) Similarly, from the latter follows C.

Alternative user utterances with underspecification:

(a) From this follows D since C implies D by Lemma Y.

— *this may refer to (1)+(2)+(3), to (3), or even (1) or (2) with wrong justification*

(b) It holds D since C implies D by Lemma Y.

— *no underspecified anaphoric reference but ambiguity at domain reasoning level*

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

UNIVERSITÄT DES SAARLANDES

Example: Proof Step Evaluation

ΩMEGA

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.

UNIVERSITÄT DES SAARLANDES

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

What are the right provers?

ΩMEGA

???

DIALOG project is (from a cognitive science perspective) a

→ very fascinating, novel application domain for deduction systems.

UNIVERSITÄT DES SAARLANDES