



Proof Development and Maintenance

(after a brief survey of the new ΩMEGA System)

Serge Autexier

`serge@ags.uni-sb.de`

DFKI GmbH & CS Department, Saarland University, Saarbrücken, Germany

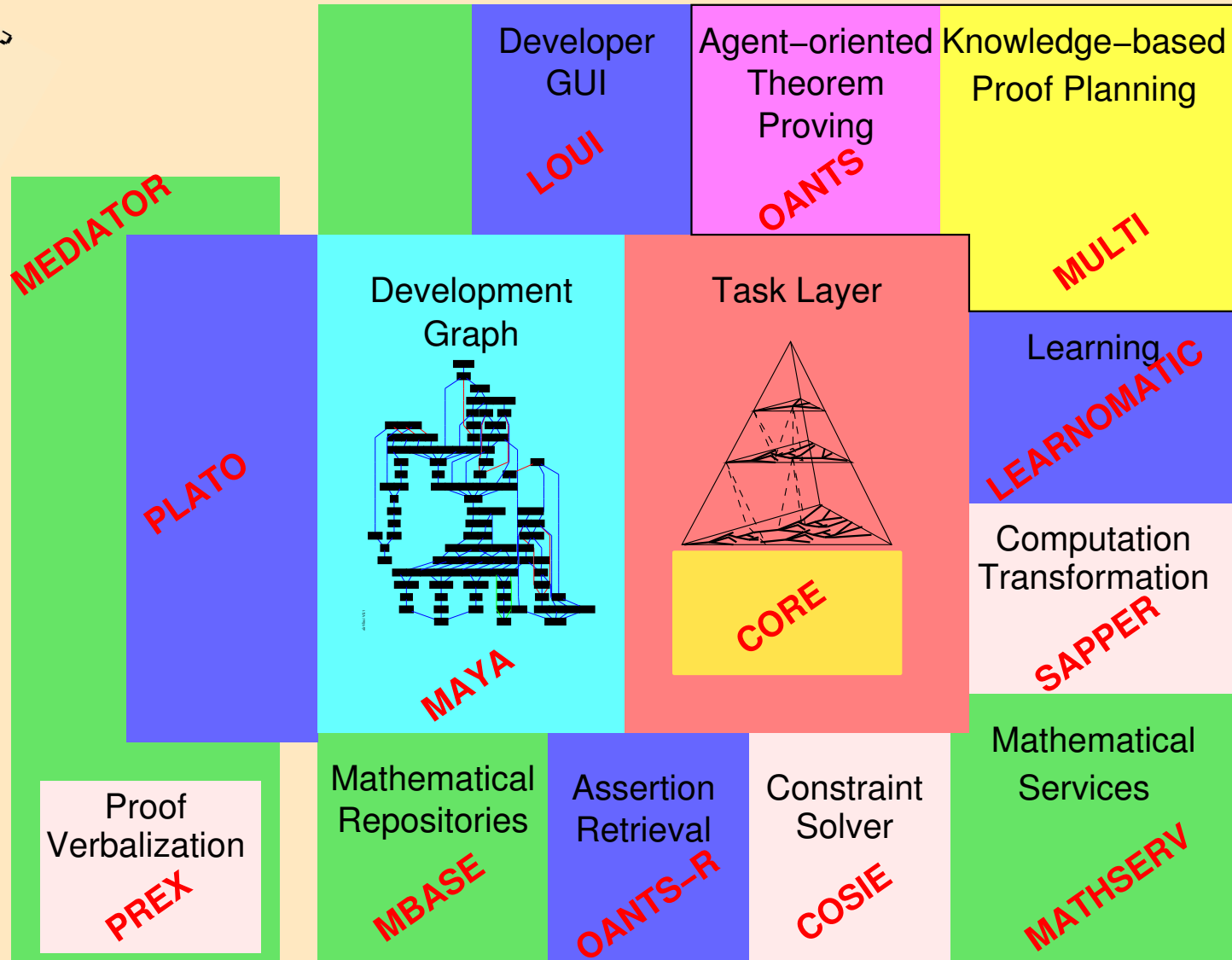
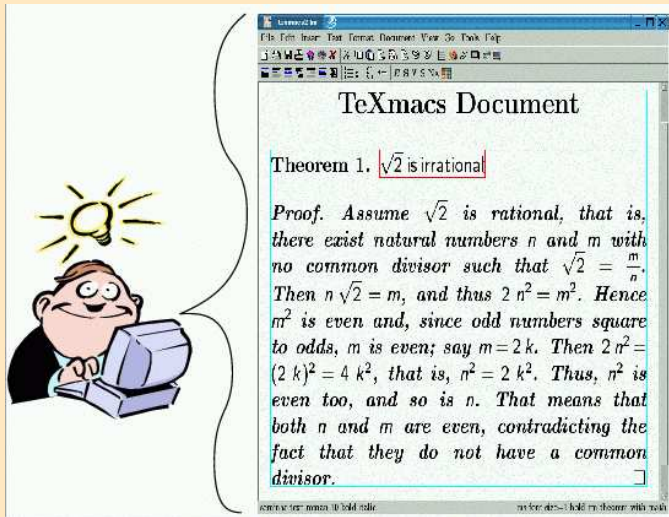
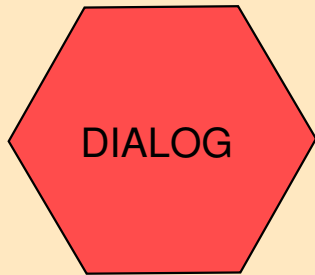
SFB-378, C-Tag, March 17, 2006

Saarbrücken, Germany



Architecture of the new OMEGA system

New Architecture



People

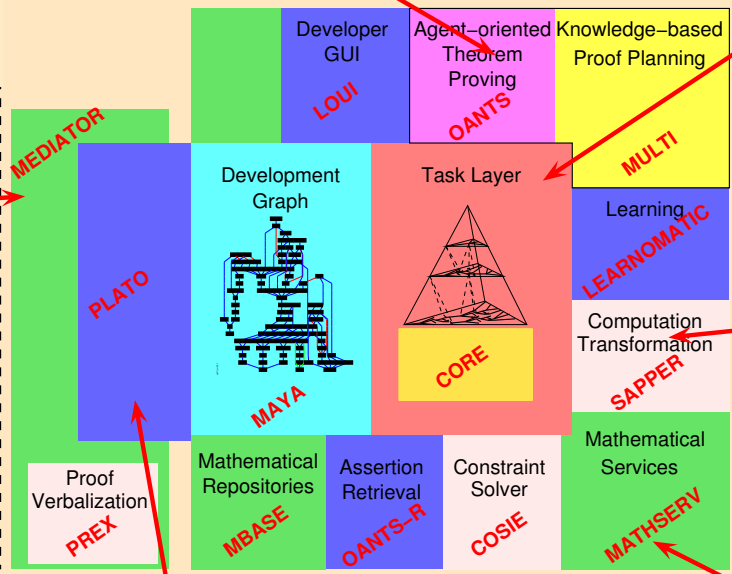
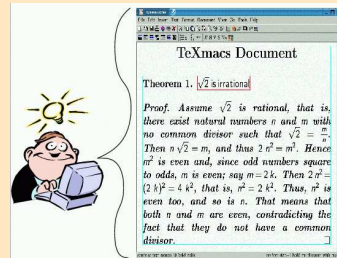


Chad Brown

(Martin Pollet)

Dominik Dietrich

Armin Fiedler
(VeriMathDoc)



Marc Wagner

Jürgen Zimmer

Frank Theiß

Andreas Franke (Software Engineer for the whole project)

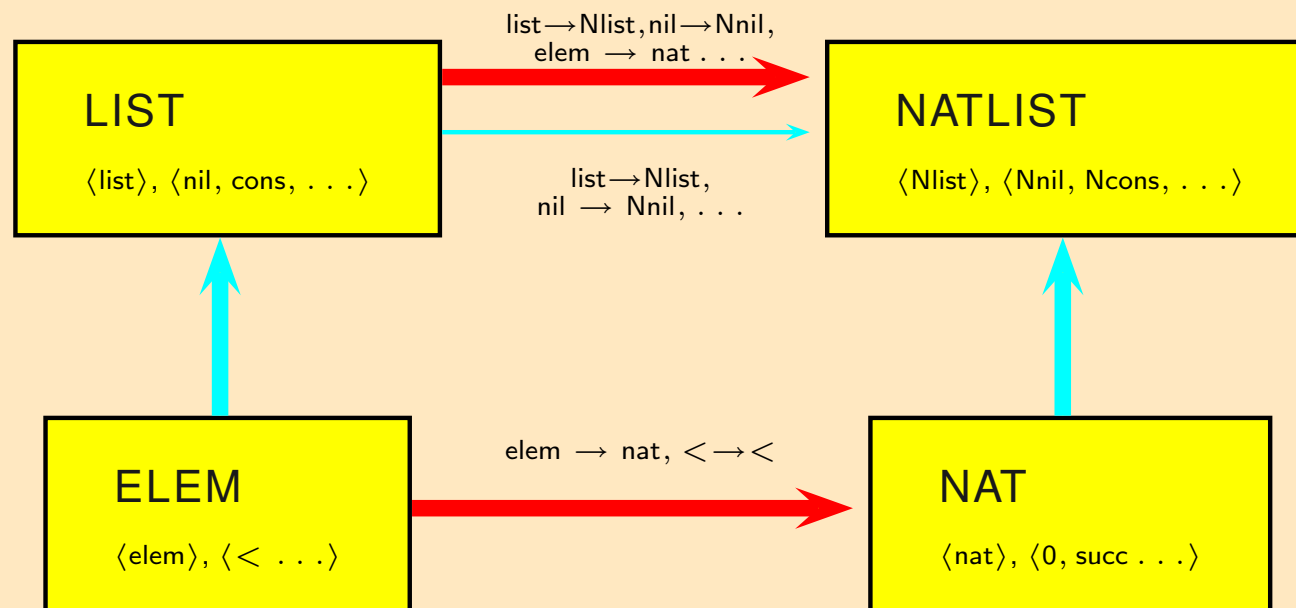
Development graphs: Maintaining structured mathematical theories

Development Graph: Basic Concepts



- Global links from N to M import complete signature and axioms from N
- Local links import local signature and axioms only

Used to represent instantiation of parameterized specifications



[Hutter'00], [AutexierHutter'02&'05], [MossakowskiAutexierHutter'01&'06]

What are Development Graphs Composed of?



Development Graph

=

Structured Logical Content of Specifications

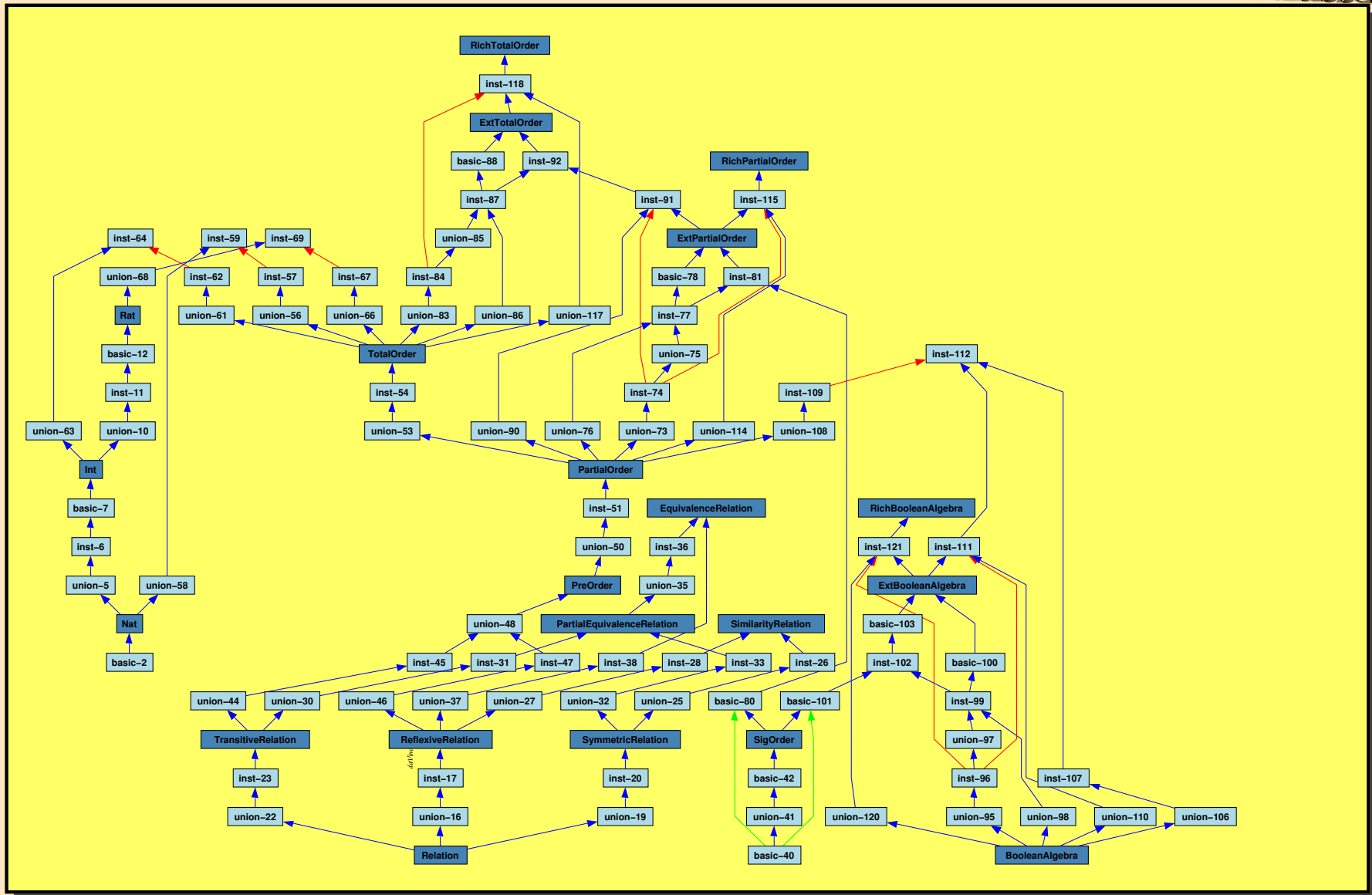
+

Status of proof obligations

(pending, proven, used axioms, ...)

Structured Specification

*Verification +
Management*



The Task Layer:

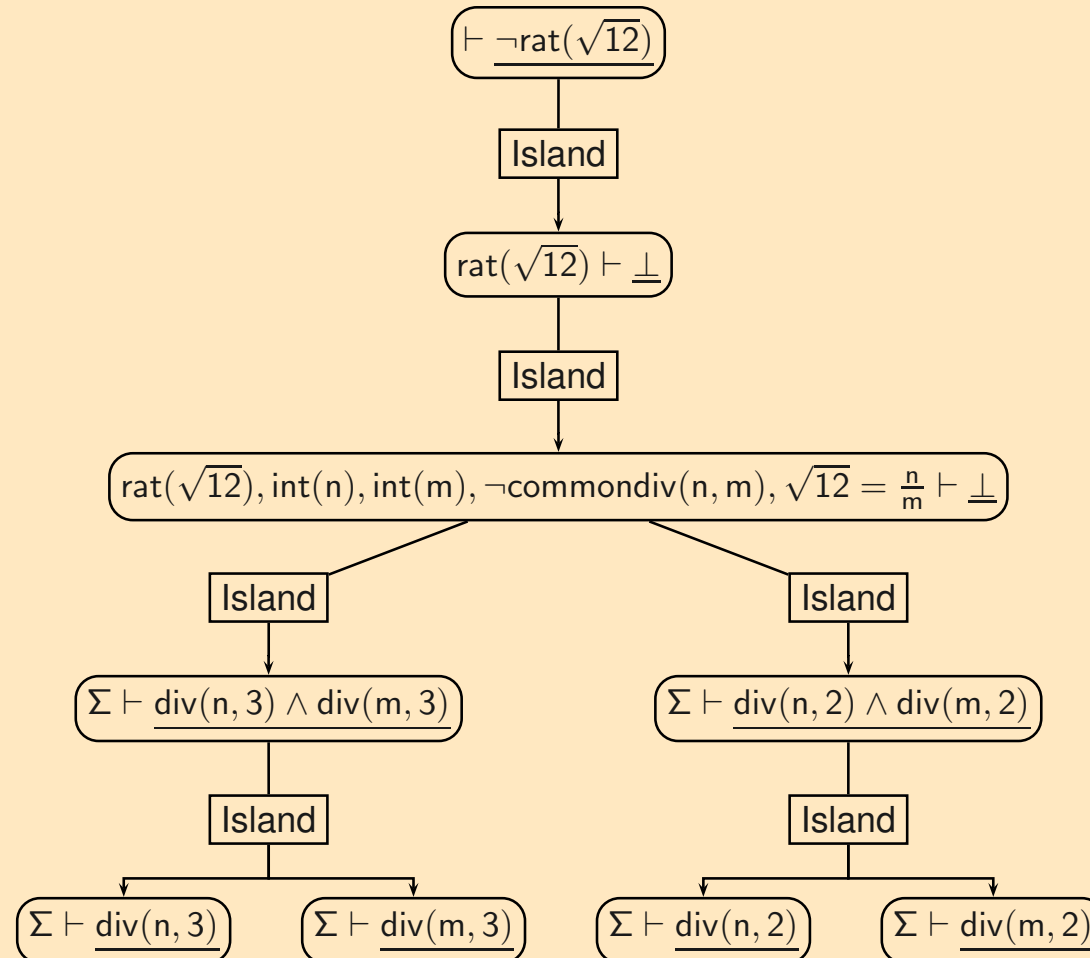
Developing and maintaining proofs, proof plans, proof sketches, and alternatives

Proof DataStructure (PDS)



- Proof simultaneously at different **levels of granularity**
 - ▶ Representation of abstract proof ideas and their refinement
(Proof Planning)
 - ▶ Representation of external systems proofs/computations and their refinement
 - ▶ Definable level of granularity (slices through the hierarchy)
Views
 - Interactive proof development
 - Adaptive natural language proof explanations
 - ▶ Allows to postpone verification (expansion) of higher-level proof steps
- **Alternative proof attempts (on the same level of granularity)**
- **Support for lemmatization (forest of PDS trees with links)**

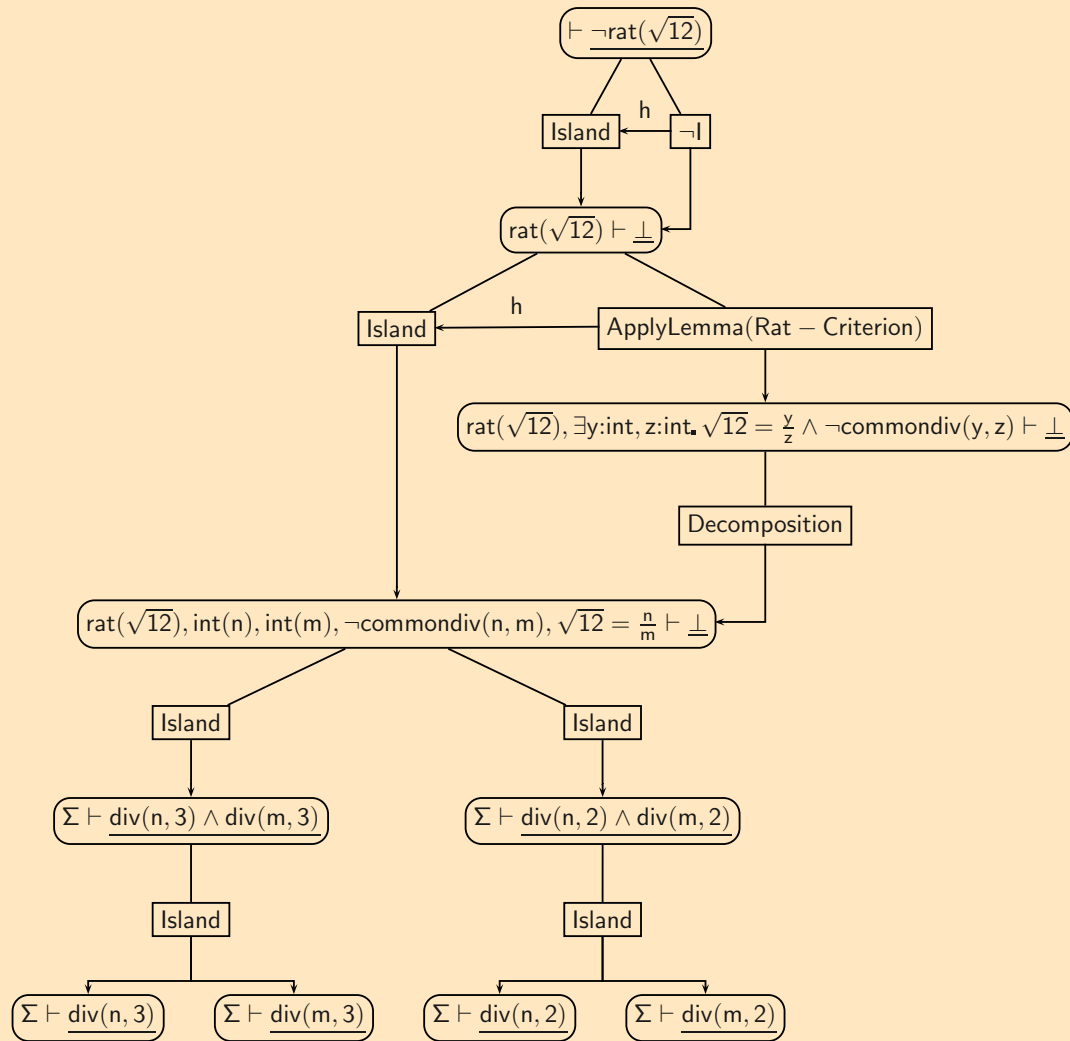
Example Abstract PDS



Example Complete PDS



Complete PDS

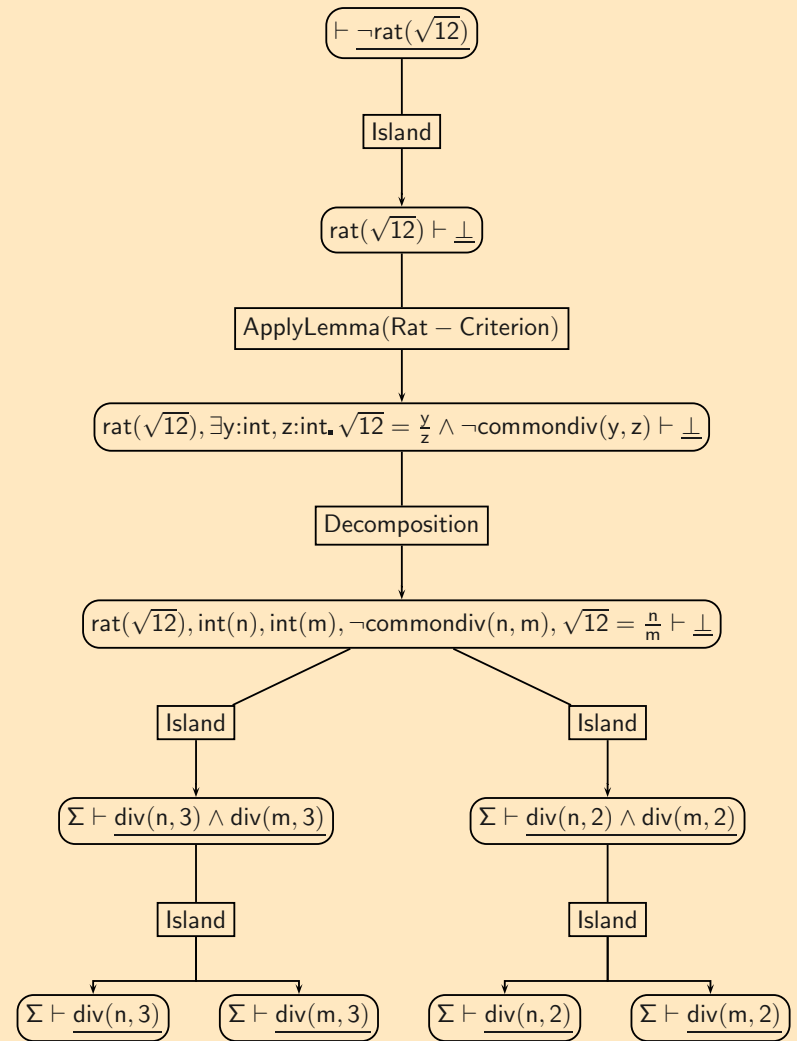
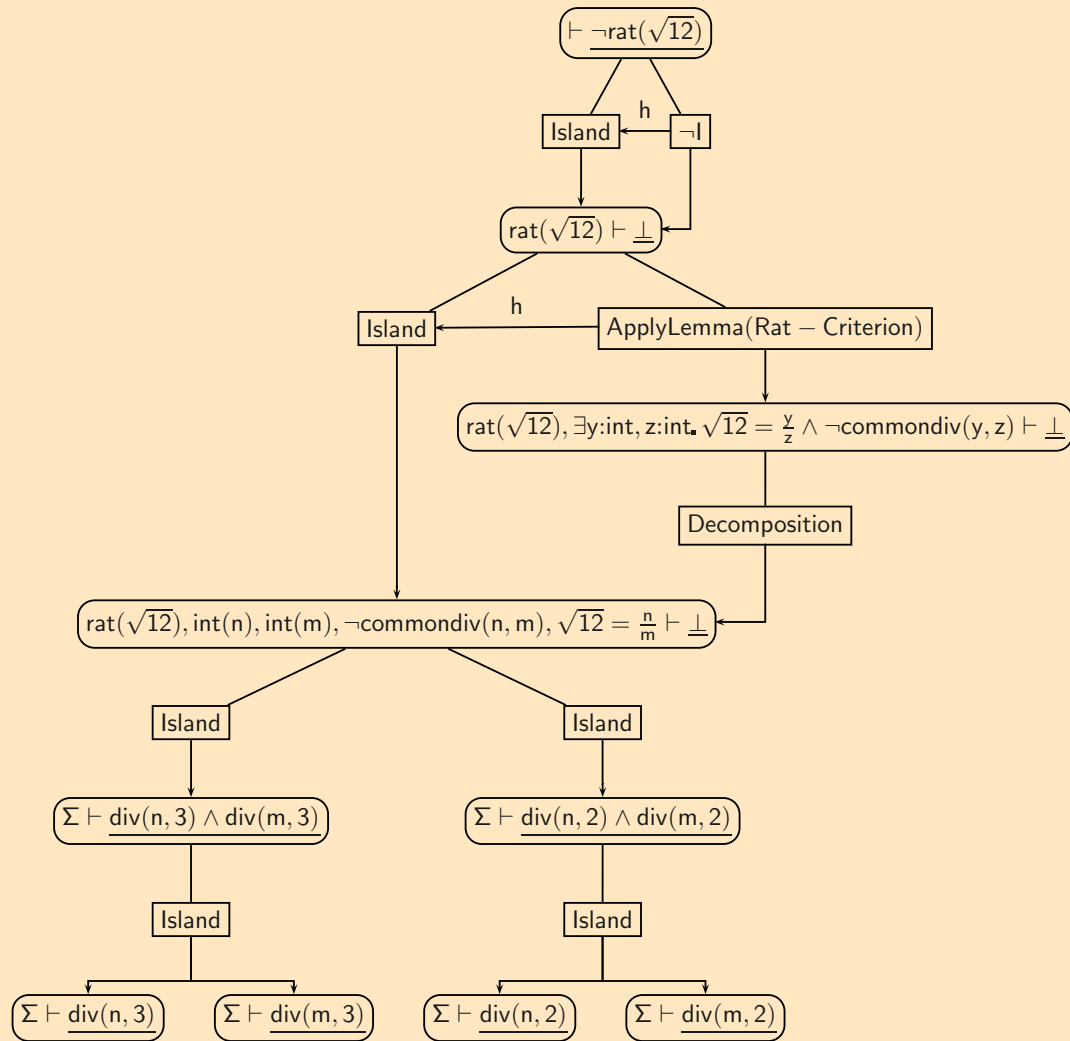


Example Complete PDS



Complete PDS

A PDS View



PDSs and Tasks

- Work on the PDS provided basis to use $\bar{\lambda}\mu\tilde{\mu}$ -calculus proof terms as semantics for OMDOC proofs extended by alternatives Semantics for OMDOC with alternatives (= PDS)

($\bar{\lambda}\mu\tilde{\mu}$ -calculus proposed by Curien and Herbelin, Curry-Howard Isomorphism to classical SK, allows call-by-value/call-by-name)

- Tasks =

multi-conclusion sequents $\varphi_1, \dots, \varphi_n \vdash \psi_1, \dots, \psi_m$

+

focuses of attention on subformulas, maintained during the proof

Alternative Substitutions

Alternative substitutions handled by *agendas* and *dependencies*

- Agenda: $\langle \{n_1, \dots, n_k\}, \{\sigma_1, \dots, \sigma_l\} \rangle$
Spanning set of tasks that are pairwise not in alternative proofs
+ Set of substitutions active on these nodes
- Dependencies: $j \leftarrow j'$
partial order among justifications that introduce substitutions

Step 0



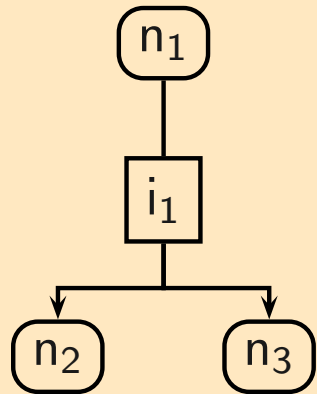
Agendas $\{\langle\{n_1\}; \{\}\rangle\}$

Dependencies $\{\}$

n_1

active substitutions
$n_1 : \emptyset$

Step 1



Agendas $\{\langle\{n_2, n_3\}; \{\sigma_{i_1}\}\rangle\}$

Dependencies $\{\}$

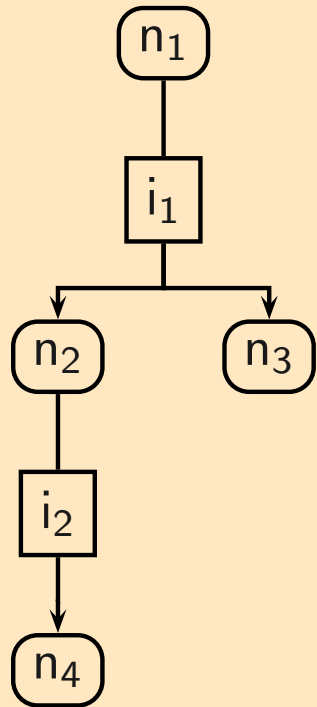
active substitutions

$n_1 : \emptyset$

$n_2 : \sigma_{i_1}$

$n_3 : \sigma_{i_1}$

Step 2



Agendas $\{\langle\{n_3, n_4\}; \{\sigma_{i_1}, \sigma_{i_2}\}\rangle\}$

Dependencies $\{i_2 \rightarrow i_1\}$

active substitutions

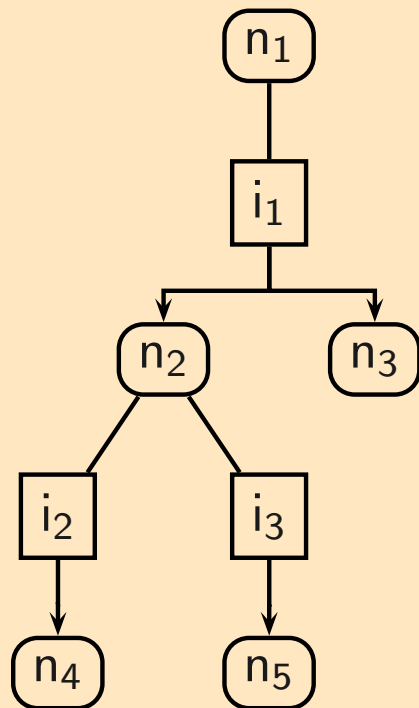
$n_1 : \emptyset$

$n_2 : \sigma_{i_1}$

$n_3 : \sigma_{i_1} \circ \sigma_{i_2}$

$n_4 : \sigma_{i_1} \circ \sigma_{i_2}$

Step 3



Agendas $\{\langle\{n_3, n_4\}; \{\sigma_{i_1}, \sigma_{i_2}\}\rangle,$

$\langle\{n_3, n_5\}; \{\sigma_{i_1}, \sigma_{i_3}\}\rangle\}$

Dependencies $\{i_2 \rightarrow i_1, i_3 \rightarrow i_1\}$

active substitutions

$n_1 : \emptyset$

$n_2 : \sigma_{i_1}$

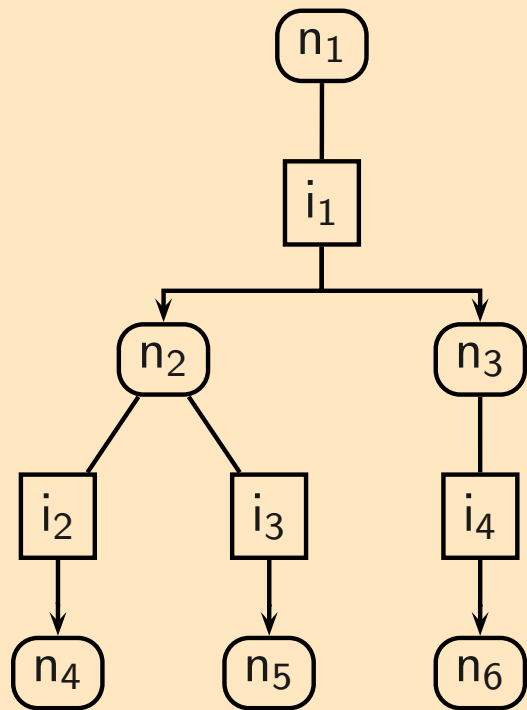
$n_3 : \sigma_{i_1} \circ \sigma_{i_2}$ for agenda $\langle\{n_3, n_4\}; \{i_1, i_2\}\rangle$

$n_3 : \sigma_{i_1} \circ \sigma_{i_3}$ for agenda $\langle\{n_3, n_5\}; \{i_1, i_3\}\rangle$

$n_4 : \sigma_{i_1} \circ \sigma_{i_2}$

$n_5 : \sigma_{i_1} \circ \sigma_{i_3}$

Step 4



Agendas $\{\langle\{n_4, n_6\}; \{\sigma_{i_1}, \sigma_{i_2}, \sigma_{i_4}\}\rangle,$

$\langle\{n_3, n_5\}; \{\sigma_{i_1}, \sigma_{i_3}\}\rangle\}$

Dependencies $\{i_2 \rightarrow i_1, i_3 \rightarrow i_1, i_4 \rightarrow i_1, i_4 \rightarrow i_2\}$

substitutions

$n_1 : \emptyset$

$n_2 : \sigma_{i_1}$

$n_3 : \sigma_{i_1} \circ \sigma_{i_2}$ for agenda $\langle\{n_4, n_6\}; \{\sigma_{i_1}, \sigma_{i_2}, \sigma_{i_4}\}\rangle$

$\sigma_{i_1} \circ \sigma_{i_3}$ for agenda $\langle\{n_3, n_5\}; \{\sigma_{i_1}, \sigma_{i_3}\}\rangle$

$n_4 : \sigma_{i_1} \circ \sigma_{i_2} \circ \sigma_{i_4}$

$n_5 : \sigma_{i_1} \circ \sigma_{i_3}$

$n_6 : \sigma_{i_1} \circ \sigma_{i_2} \circ \sigma_{i_4}$



Inferences

Examples of Inferences



$$\frac{P_1 : A \subseteq B \quad P_2 : B \subseteq C}{C : A \subseteq C}$$

Application Condition: —

Examples of Inferences



$$\begin{array}{c} [\epsilon > 0, D > 0, 0 < |x - A|, |x - A| < D] \\ \vdots \\ P : |F(x) - L| < \epsilon \\ \hline C : \lim_A F = L \end{array} \quad \textit{Limit}(\epsilon, x)$$

Application Condition: $EV(\epsilon, \{F, A, L\}) \wedge EV(x, \{F, A, L, D\})$

Examples of Inferences



$$\frac{P_1 : F \quad P_2 : U = V}{C : G} = \text{subst-}m(\pi)$$

Application Condition: –

Outline functions:

$\langle C \quad \text{compute-}=\text{subst-prem}(P_1, P_2, \pi) \rangle$

$\langle P_1 \quad \text{compute-}=\text{subst-prem}(C, P_2, \pi) \rangle$

There were 4 tactics and 2 methods in the old ΩMEGA system!

Application of Inferences



- Inference are applied on subformulas of a task
- Not all premises and conclusions need to be mapped.
- Premises are mapped to negative positions in a task
- Conclusions are mapped to positive positions in a task
- Example:

$$P \Rightarrow (A \subset B) \vdash Q \Rightarrow (A \subset C) \quad \frac{P_1 : (A \subset B) \quad P_2 : (B \subset C)}{C : (A \subset C)} \text{Subset}$$

Suppose we map $P_1 \mapsto \langle 1, 2 \rangle$ and $C \mapsto \langle 2, 2 \rangle$ (Hence $P_2 \rightarrow B \subset C$).

$$P \Rightarrow (A \subset B) \vdash Q \Rightarrow (P \wedge (B \subset C))$$

Interesting Questions



$$\frac{P_1 : F \quad P_2 : U = V}{C : G} =_{\text{subst-m}(\pi)}$$

- When is an inference applicable?

Application Condition: –

Outline functions:

$\langle C \text{ compute} == \text{subst-prem}(P_1, P_2, \pi) \rangle$

$\langle P_1 \text{ compute} == \text{subst-prem}(C, P_2, \pi) \rangle$

As soon as we have matched enough premises and conclusions such that we can compute any missing premise and conclusion using the outline functions

Application directions = These sets of premises and conclusions
 \implies Use for proof planning instead of methods

- Which agents do we need?

An Ω_{ANTS} -argument agent $\langle \mathcal{G}, \mathcal{D} \rangle$ searches

- ▶ for possible instantiations for premises/conclusions (*goal set \mathcal{G}*)
- ▶ subject to existing instantiations for other premises/conclusions

(*dependency set \mathcal{D}*)

Agent Creation Graph



$$\frac{P_1 : F \quad P_2 : G}{C : H}$$

Application Condition: $P(P_1, P_2, C)$

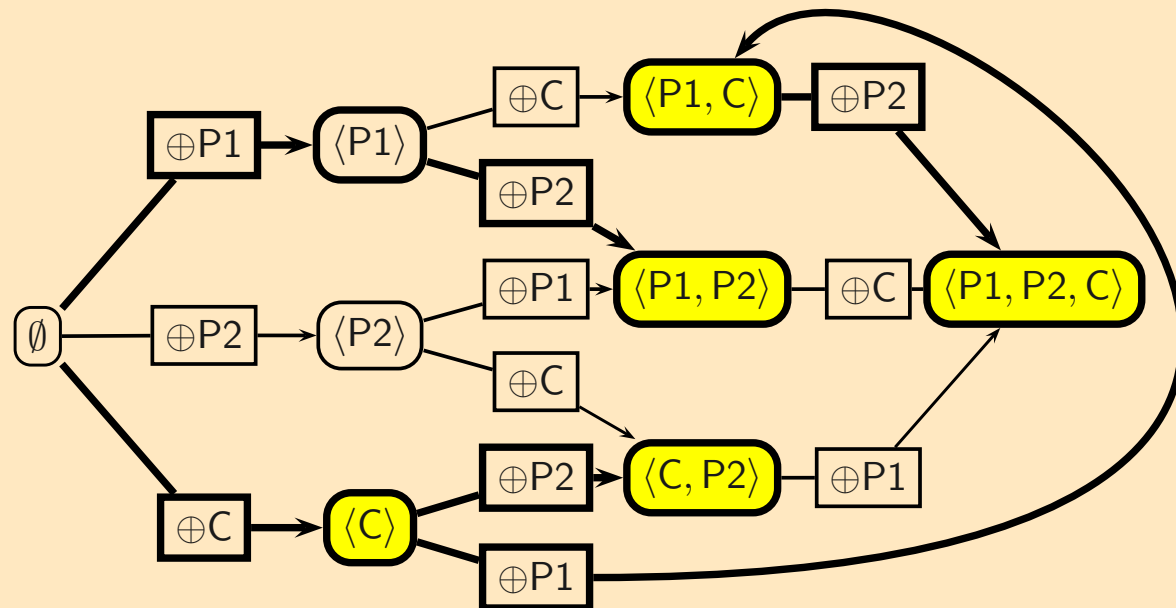
Outline functions: $\langle P_2 \ a(C) \rangle$,
 $\langle P_1 \ b(P_2, C) \rangle$,
 $\langle C \ c(P_1, P_2) \rangle$

Application directions:

$\langle C \rangle$, $\langle C, P_2 \rangle$, $\langle P_1, P_2 \rangle$, $\langle P_1, C \rangle$, and
 $\langle P_1, P_2, C \rangle$.

Each edge

$\mathcal{D} \rightarrow (\mathcal{D} \cup \{G\})$ is an
 atomic agent $\langle \mathcal{D}, \{G\} \rangle$



Next Steps



- Automation of proof search at the Tasklayer
 - ▶ Reactivate Ω_{ANTS} on that basis
 - ▶ Reimplement proof planner `MULTI`
- Further evaluation:
 - ▶ Coding effort is already drastically reduced
 - ▶ Less inferences required by deep application of inferences
 - ▶ Benefit for automated proof search?
- Specific services required to support proof development in `TEXMACS` via `PLAT Ω`

Further Details

- PDS (AutexierBenzmüllerDietrichMeierWirth,MKM 2005)
Basis to use $\bar{\lambda}\mu\tilde{\mu}$ -calculus proof terms as semantics for OMDOC proofs
extended by alternatives (Autexier&Sacerdoti-Coen, Submitted)

- Tasklayer: (Diploma Thesis Dietrich)
 - ▶ Handling alternative substitutions
 - ▶ Inferences, Application directions and Generating argument agents
+ synthesizing inferences from Math. Theories
(Autexier&Dietrich, Submitted)