Underspecification in Math-DIALOG

Christoph Benzmüller

DIALOG Project

Computer Science & Comput. Ling.
Saarland University
Saarbrücken, Germany
http://www.ags.uni-sb.de/~dialog/

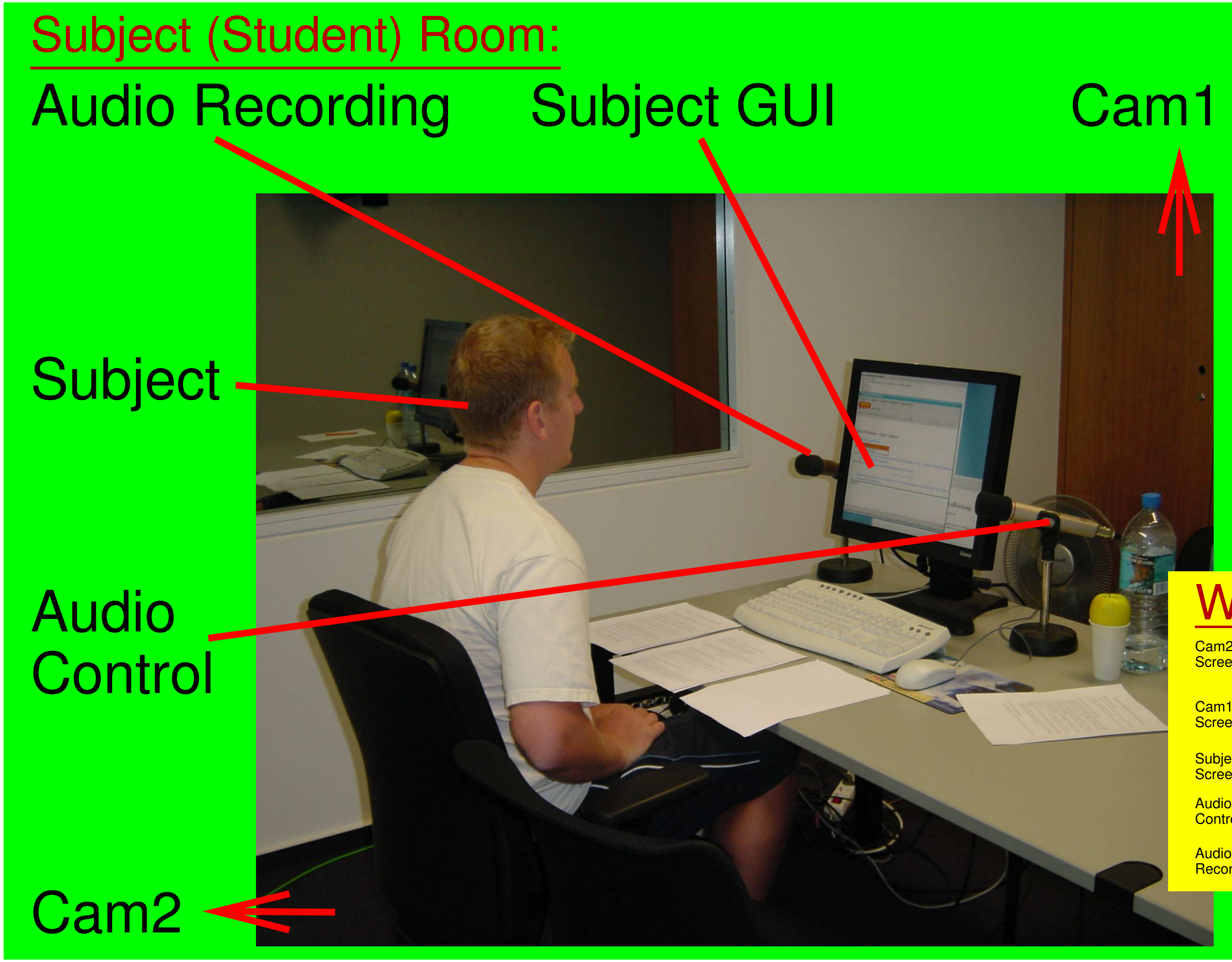
C-Tag, March 17, Saarbrücken, Germany

Motivation and Goal of the Project

Assume that $a \in X$.
If $X \cap Y = \emptyset$,
then $a \notin Y$.

well done!

WOZ-Experiment → Own Corpus



WOZ-Experiment → Own Corpus

DIALOG Project

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

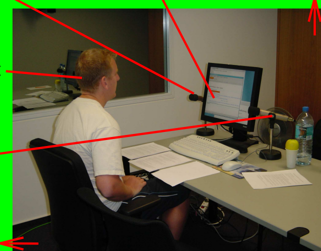
Subject GUI

Cam1

Subject

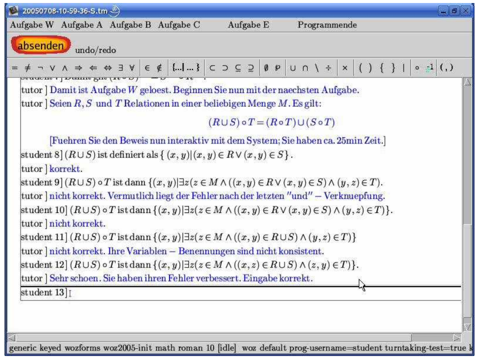
Audio Control

Cam2

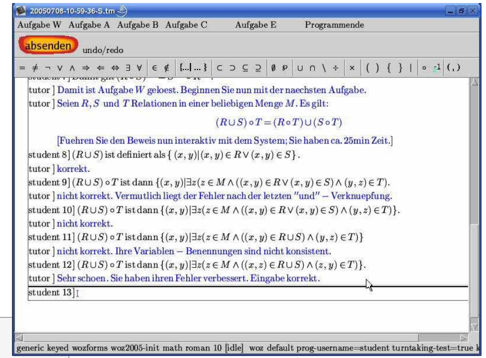


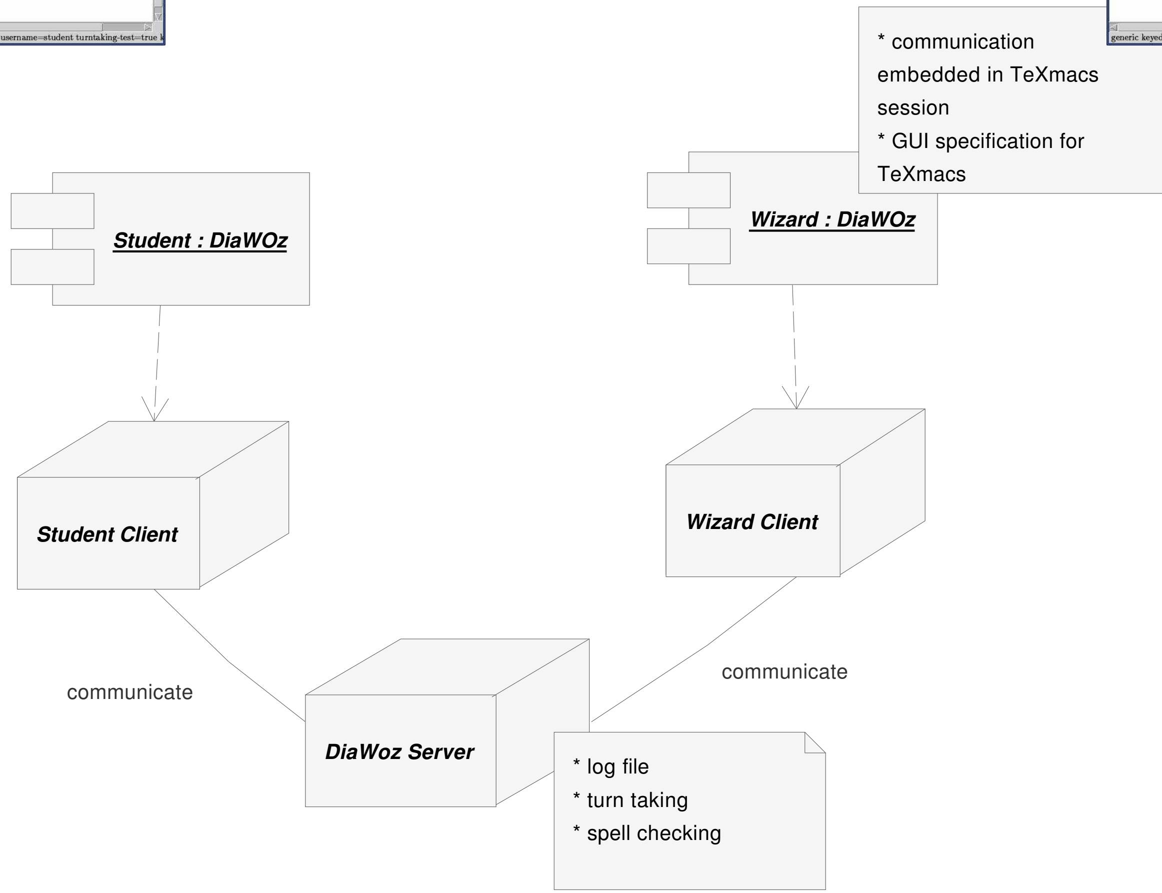
Architecture of DIAWOZ-II

DIALOG Project



More than a configurable chat tool for maths . . .





UNIVERSITÄT DES SAARLANDES

Corpus Example: 1. Experiment (2003)

DIALOG Project

T1: Bitte zeigen Sie: $K((A \cup B) \cap (C \cup D)) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$!
[Please show: $K((A \cup B) \cap (C \cup D)) = (K(A) \cap K(B)) \cup (K(C) \cap K(D))$!]

S1: nach deMorgan-Regel-2 ist $K((A \cup B) \cap (C \cup D)) = (K(A \cup B) \cup K(C \cup D))$.
[by deMorgan-Rule-2 $K((A \cup B) \cap (C \cup D)) = (K(A \cup B) \cup K(C \cup D))$ holds.]

T2: Das ist richtig!
[This is correct!]

S2: $K(A \cup B)$ ist laut deMorgan-1 $K(A) \cap K(B)$
[$K(A \cup B)$ is $K(A) \cap K(B)$ according to deMorgan-1]

T3: Das stimmt auch.
[That is also right.]

S3: und $K(C \cup D)$ ist ebenfalls laut deMorgan-1 $K(C) \cap K(D)$
[and $K(C \cup D)$ is also $K(C) \cap K(D)$ according to deMorgan-1]

...

Corpus Example: 2. Experiment (2005)

DIALOG Project

S0: was ist $\circ$

T0: Das Relationenprodukt, auch Komposition von Relationen genannt. Bitte schauen Sie sich die Definition unter Abschnitt 4 an. `(k.A.; k.A.; k.A.)`

S1: $(R\circ S)^{-1} = \{(z,x) | \exists y((x,y) \in R \wedge (y,z) \in S\}$

T1: Das ist korrekt. `(korrekt; angemessen; relevant)`

S2: $R^{-1} = \{(x,y)|(y,x) \in R\}$

T2: Ebenfalls korrekt. `(korrekt; angemessen; relevant)`

S3: Also ist $S^{-1} \circ R^{-1} = \{(v,x)| v \in S^{-1} \wedge x \in R^{-1}\}$

T3: Nein. Auch die inversen Relationen, S^{-1} und R^{-1}, sind binaere Relationen! `(inkorrekt; angemessen; relevant)`

S4: Also ist $S^{-1} \circ R^{-1} = \{(v,x)| \exists z((v,z) \in S^{-1} \wedge (z,x) \in R^{-1}\}$

T4: ...

Different Corpora

DIALOG Project

Observation in 2005 corpus:

- Two dialog fragments for exercise W: $(R \circ S)^{-1} = S^{-1} \circ R^{-1}$
- Student A

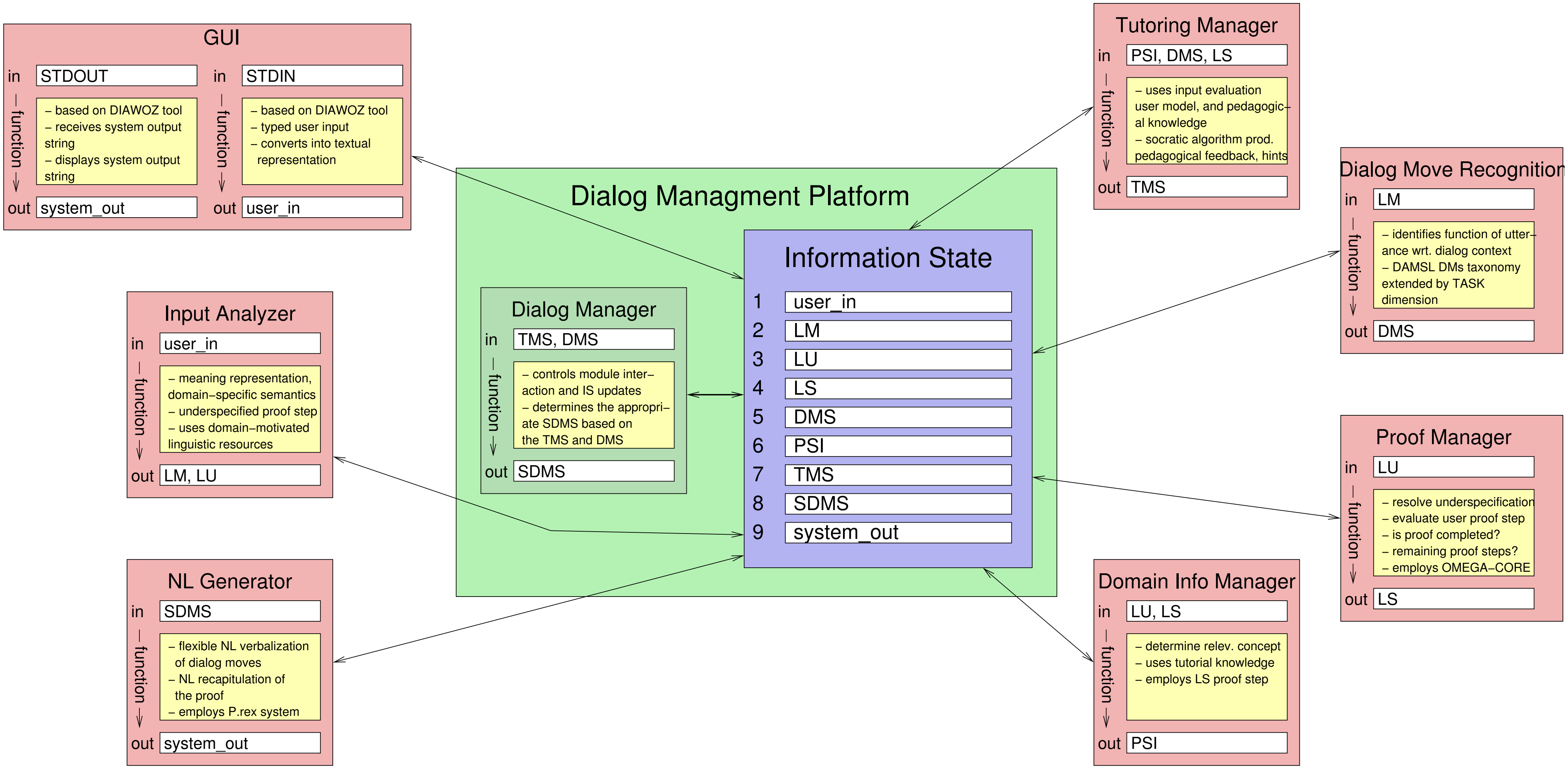
student]	One needs to show equality between two sets
tutor]	That's right! How do you proceed?
student]	I use the extensionality principle
tutor]	That's correct!

- Student B

student]	$(R \circ S)^{-1} = \{(x, y) \mid (y, x) \in (R \circ S)\}$
tutor]	correct
student]	$(R \circ S)^{-1} = \{(x, y) \mid (y, x) \in \{(x, y) \mid \exists z(z \in M \wedge (x, z) \in R \wedge (z, y) \in S\}\}$
tutor]	okay, but can be done simpler.

The DIALOG System and Components

DIALOG Project



UNIVERSITÄT DES SAARLANDES

Challenge: Mathural Analysis

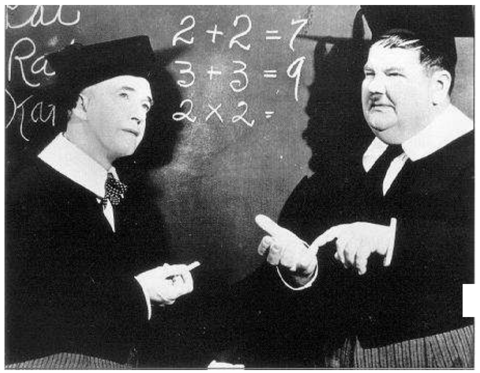
Challenge: Proof Step Evaluation

DIALOG Project

Perspective of Mathematical Domain Reasoning (MDR):

- Support for resolution of Ambiguiti[illegible]d Underspecification
- Proof Step Evaluation
  - Soundness: proof[illegible]erifiable by formal system?
  - Granularity[illegible]rgumentative complexity of proof step?
  - Relevanc[illegible] proof step needed/useful in achieving the goal?

Logical vs Tutorial Dimension



—— declarative abstract level sketches ➤

Communication Gap

⟵ procedural calculus level proofs ——

MDR as a basis for Tutoring

Notion of Underspecification

(DM-1) ...
(DM-2) ...
?
(G) ...

Given: (DM-1) $\overline{X \cup Y} = \overline{X} \cap \overline{Y}$

(DM-2) $\overline{X \cap Y} = \overline{X} \cup \overline{Y}$

Task: Please show $\overline{(A \cup B) \cap (C \cup D)} = (\overline{A} \cap \overline{B}) \cup (\overline{C} \cap \overline{D})$

New: By deMorgan $\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$.

Semantics in Math-DIALOG

DIALOG Project

Reading 1:

$\boxed{\text{By}}$ instantiation of $\boxed{\text{DM}}$-2 where we choose $A \cup B$ for x and $C \cup D$ for y we get $\boxed{\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}}$

DM-1	$\forall x, y.\ \overline{x \cup y} = \bar{x} \cap \bar{y}$	Hyp/Given
DM-2	$\forall x, y.\ \overline{x \cap y} = \bar{x} \cup \bar{y}$	Hyp/Given
Intermediate-1	$\forall y.\ \overline{(A \cup B) \cap y} = \overline{(A \cup B)} \cup \bar{y}$	$\forall$-Inst(DM-2, $(A \cup B)$)
New	$\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$	$\forall$-Inst(Intermed., $(C \cup D)$)
	¿	
Task	$\overline{(A \cup B) \cap (C \cup D)} = (\bar{A} \cap \bar{B}) \cup (\bar{C} \cap \bar{D})$	Open

Semantics in Math-DIALOG

DIALOG Project

Reading 2:

By applying DM-1 twice to our proof task we can reduce/rewrite the task to

first from right to left with $X \leftarrow A$ and $y \leftarrow B$ and
second from right to left with $X \leftarrow C$ and $y \leftarrow D$ at
positions [2.1] and [2.2] resp.

$$\boxed{\overline{(A\cup B)\cup(C\cup D)} = \overline{(A\cup B)} \cup \overline{(C\cup D)}}$$

Lots of variations possible:
- step-order
- Subst

	DM-1	$\forall x,y.\ \overline{x\cup y} = \bar{x}\cap\bar{y}$	Hyp/Given
	DM-2	$\forall x,y.\ \overline{x\cap y} = \bar{x}\cup\bar{y}$	Hyp/Given
1	Intermediate-1	$\forall y.\ \overline{A\cup y} = \bar{A}\cap\bar{y}$	∀-Inst (DM-1, A)
2	Intermediate-2	$\overline{A\cup B} = \bar{A}\cap\bar{B}$	∀-Inst (Intermed-1, B)
4	Intermediate-~~1~~4	$\forall y.\ \overline{C\cup y} = \bar{C}\cap\bar{y}$	∀-Inst (DM-1, C)
5	Intermediate-5	$\overline{C\cup D} = \bar{C}\cap\bar{D}$	∀-Inst (Intermed.-4, D)
		?	
6	New	$\overline{(A\cup B)\cap(C\cup D)} = \overline{(A\cup B)}\cup\overline{(C\cup D)}$	Open
3	Intermediate-3	$\overline{(A\cup B)\cap(C\cup D)} = \overline{(A\cup B)}\cup(\bar{C}\cap\bar{D})$	=Subst([2.2], Intermed.-5)
	Task	$\overline{(A\cup B)\cap(C\cup D)} = (\bar{A}\cap\bar{B})\cup(\bar{C}\cap\bar{D})$	=Subst ([2.1], Intermed.-2)

Semantics in Math-DIALOG

DIALOG Project

Reading-1 :

By instantiation of DM-2 with $A \cup B$ for x and $C \cup D$ for y
we get $\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$

DM-1	$\forall x y .\ \overline{x \cup y} = \bar{x} \cap \bar{y}$	Hyp / Given
DM-2	$\forall x y .\ \overline{x \cap y} = \bar{x} \cup \bar{y}$	Hyp / Given
New	$\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$	$\dfrac{\forall\text{-Inst* (DM-2, (A}\cup\text{B), (}}{\text{Assertion (DM-2)}}$
	?	
Task	$\overline{(A \cup B) \cap (C \cup D)} = (\bar{A} \cap \bar{B}) \cup (\bar{C} \cap \bar{D})$	Open

Semantics in Math-DIALOG

Reading-2

$\boxed{\text{By}}$ applying $\boxed{\text{DM-1}}$ we can reduce / rewrite the task to $\boxed{\overline{(A\cup B)\cup(C\cup D)} = \overline{(A\cup B)}\cup\overline{(C\cup D)}}$

DM-1	$\forall x y.\ \overline{x\cup y} = \bar{x}\cap\bar{y}$	Hyp/Given
DM-2	$\forall x y.\ \overline{x\cap y} = \bar{x}\cup\bar{y}$	Hyp/Given

2

New	$\overline{(A\cup B)\cap(C\cup D)} = \overline{(A\cup B)}\cup\overline{(C\cup D)}$	Open
Task	$\overline{(A\cup B)\cap(C\cup D)} = (\bar{A}\cap\bar{B})\cup(\bar{C}\cap\bar{D})$	$\dfrac{\text{=Subst* (DM-1, [2.1], [2.2], ...)}}{\text{Assertion (DM-1)}}$

Semantics in Math-DIALOG

- Generic Language for Proof Objects: Encoding of Reading-1

Assume ϵ in
Fact DM-1: $\forall X, Y. \overline{(X \cup Y)} = \overline{X} \cap \overline{Y}$ by Hyp from Set-Theory;
Fact DM-2: $\forall X, Y. \overline{(X \cap Y)} = \overline{X} \cup \overline{Y}$ by Hyp from Set-Theory;
Fact New: $\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$ by Assertion from DM-2;
to obtain Task: $\overline{(A \cup B) \cap (C \cup D)} = (\overline{A} \cap \overline{B}) \cup (\overline{C} \cap \overline{D})$ by . from .; ϵ

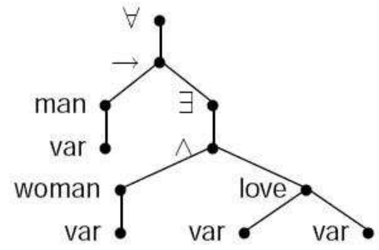
Semantics in Math-DIALOG

- Generic Language for Proof Objects: Encoding of Reading-2

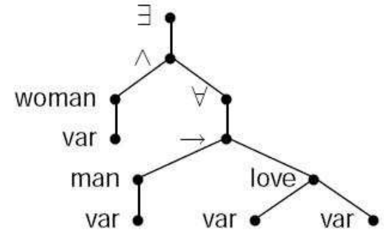
Assume ϵ in
Fact DM-1: $\forall X, Y. \overline{(X \cup Y)} = \overline{X} \cap \overline{Y}$ by Hyp from Set-Theory;
Fact DM-2: $\forall X, Y. \overline{(X \cap Y)} = \overline{X} \cup \overline{Y}$ by Hyp from Set-Theory;
Fact New: $\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$ by .
from . ;
to obtain Task: $\overline{(A \cup B) \cap (C \cup D)} = (\overline{A} \cap \overline{B}) \cup (\overline{C} \cap \overline{D})$ by
Assertion from New; ϵ

Standard Notion of Underspecification

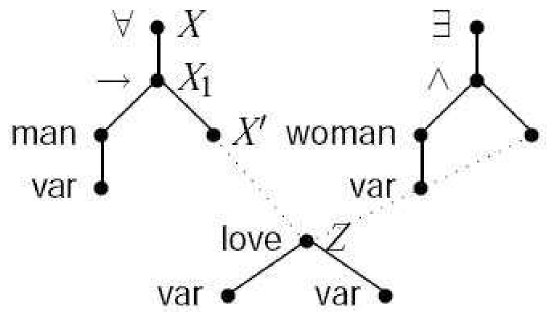
- Utterance: Every man loves a woman.
- ‘Semantics’:
  - $\forall x.\mathrm{man}(x) \longrightarrow \exists y.(\mathrm{woman}(y) \wedge \mathrm{love}(x, y))$



  - $\exists y.\mathrm{woman}(y) \wedge \forall x.(\mathrm{man}(x) \longrightarrow \mathrm{love}(x, y))$



  - . . . propably many more . . .
- A representation (+ inference support) that avoids enumeration of these readings:



Underspecification in MathDIALOG

- Utterance: By deMorgan $\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$.
- 'Semantics':
  - Partial-Proof-1 (see slide before)
  - Partial-Proof-2 (see slide before)
  - …propably many more …
- A representation (+ inference support) that avoids enumeration of these readings:

  Assume ϵ in

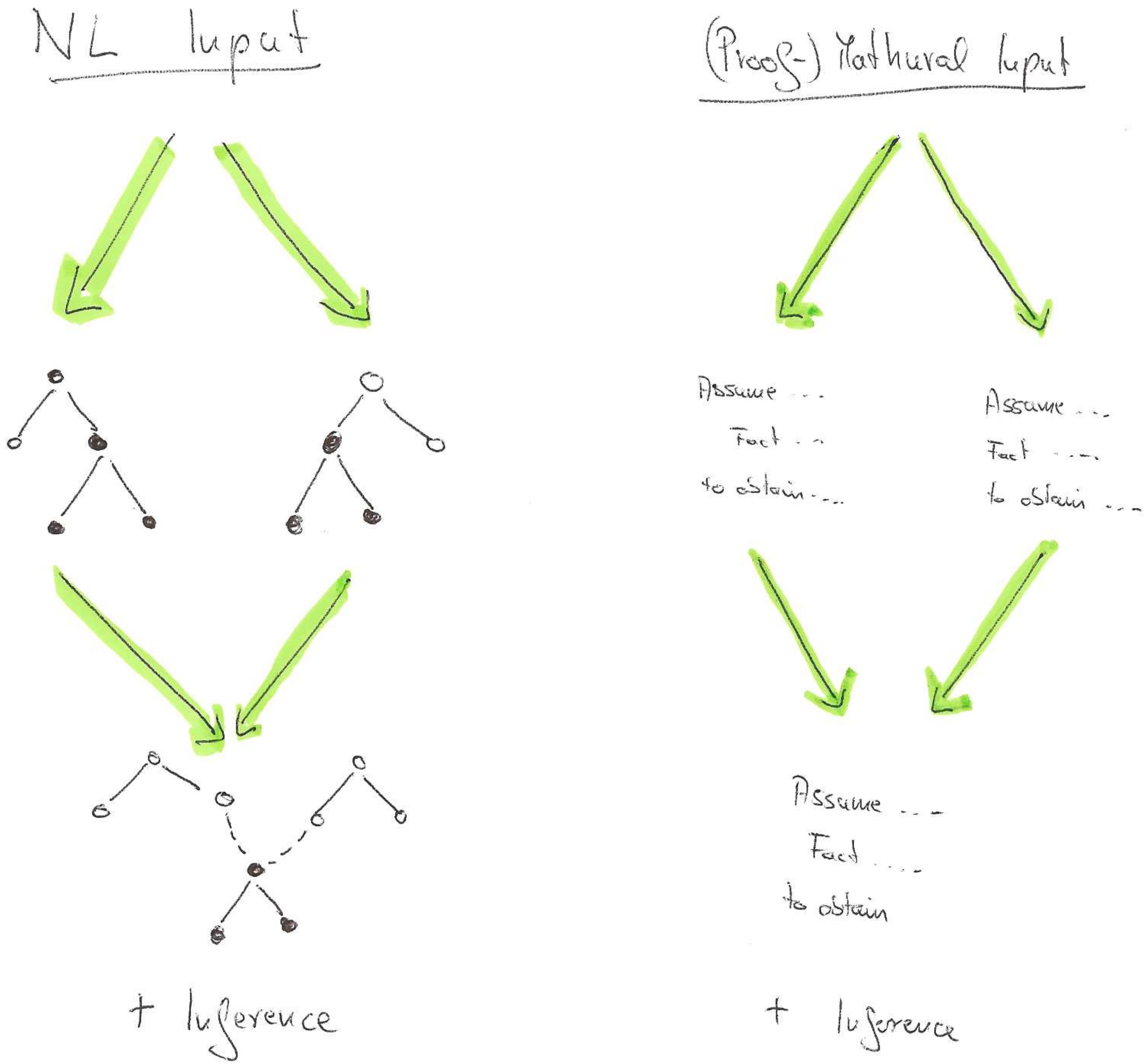
  Fact DM-1: $\forall X, Y. \overline{(X \cup Y)} = \overline{X} \cap \overline{Y}$ by Hyp from Set-Theory;

  Fact DM-2: $\forall X, Y. \overline{(X \cap Y)} = \overline{X} \cup \overline{Y}$ by Hyp from Set-Theory;

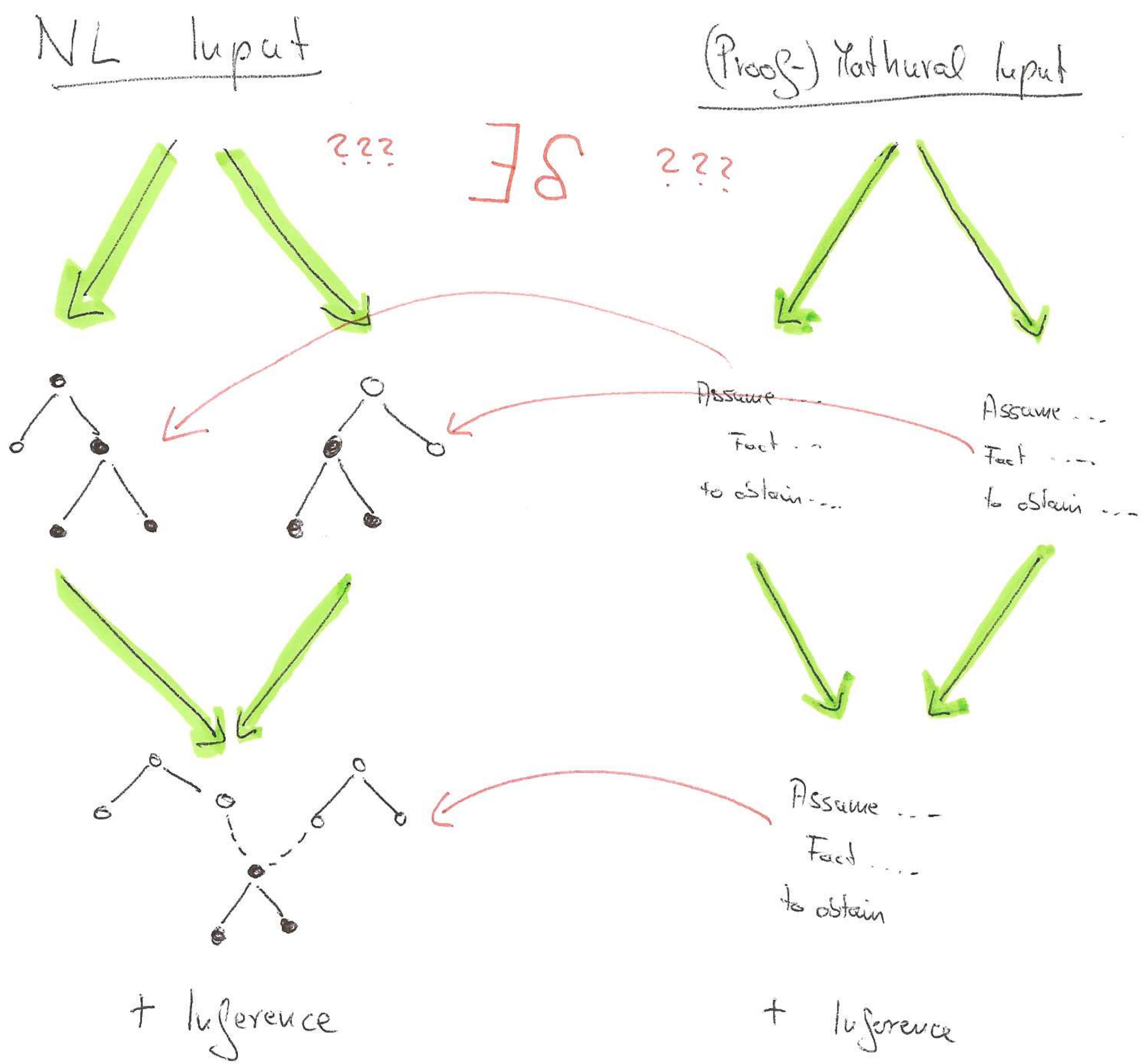
  Fact New: $\overline{(A \cup B) \cap (C \cup D)} = \overline{(A \cup B)} \cup \overline{(C \cup D)}$ by DM from .;

  to obtain Task: $\overline{(A \cup B) \cap (C \cup D)} = (\overline{A} \cap \overline{B}) \cup (\overline{C} \cap \overline{D})$ by . from .; ϵ

Are the Notions so Different?



Are the Notions so Different?



Candidates:

- Curry-Howard-Isomorphism (Propositions as Types)
- Gödelisation
- (Higher-Order) Abstract Syntax
- ... ?

Are the Notions so Different?

DIALOG Project

$$
\begin{array}{ll}
\mathtt{S} ::= \mathtt{A};\mathtt{S} \mid \mathit{Trivial} \mid \epsilon & \\
\mathtt{A} ::= \mathit{Fact}\ \ \mathtt{N:F}\ \ \mathit{by}\ \ \mathtt{R}^{*}\ \ \mathit{from}\ \ \mathtt{R}^{*} & \\
\quad \mid \mathit{Subgoals}\ \ (\mathtt{N:F})^{+}\ \ \mathit{in}\ \ \mathtt{S}^{+}\ \ \mathit{to\ obtain}\ \ \mathtt{N:F}\ \ \mathit{by}\ \ \mathtt{R}^{*}\ \ \mathit{from}\ \ \mathtt{R}^{*} & \\
\quad \mid \mathit{Assume}\ \ \mathtt{H}^{*}\ \ \mathit{in}\ \ \mathtt{S}\ \ \mathit{to\ obtain}\ \ \mathtt{N:F}\ \ \mathit{by}\ \ \mathtt{R}^{*}\ \ \mathit{from}\ \ \mathtt{R}^{*} & \\
\quad \mid \mathit{Assign}\ (\mathtt{VAR} := \mathtt{TERM} \mid \mathtt{CONST} := \mathtt{TERM}) & \\
\quad \mid \mathit{Or}(\mathtt{S} \| \ldots \| \mathtt{S}) & \\
\quad \mid \mathit{Cases}\ \mathtt{F}^{+} : (\mathit{Case}\ \mathtt{N:F:S}\ \mathit{End})^{+}\ \ \mathit{to\ obtain}\ \mathtt{N:F} & \\
\qquad \mathtt{H} ::= \mathtt{N:F} & \mathtt{CONST} ::= \mathit{const}\ \ \mathtt{N} \\
\qquad\quad \mid \mathtt{CONST:\ TYPE?} & \mathtt{VAR} ::= \mathit{var}\ \ \mathtt{N} \\
\qquad\quad \mid \mathtt{VAR:\ TYPE?} & \\
\qquad \mathtt{R} ::= (\mathtt{N,F,P}) & \mathtt{N} ::= \mathtt{STRING} \mid . \\
\qquad \mathtt{F} ::= \mathtt{FORMULA} \mid . & \mathtt{P} ::= \mathtt{POSITION} \mid .
\end{array}
$$

Translatable into Type Theory?

Are the Notions so Different?

Choose some Base Types: $\{\mathsf{o}(\text{for } \mathsf{F}), \mathsf{S}, \mathsf{A}, \mathsf{N}, \mathsf{P}, \ldots\}$

Define HOL Signature: $\epsilon : \mathsf{S}$

$\mathsf{Trivial} : \mathsf{S}$

$\_;\_ : \mathsf{A} \rightarrow \mathsf{S} \rightarrow \mathsf{S}$

$\mathsf{Fact\_by\_from} : (\mathsf{N} \rightarrow \mathsf{o} \rightarrow \mathsf{H}) \rightarrow \mathsf{List}(\mathsf{R}) \rightarrow \mathsf{List}(\mathsf{R}) \rightarrow \mathsf{A}$

$\vdots$

$\mathsf{Or} : \mathsf{List}(\mathsf{S}) \rightarrow \mathsf{A}$

$\vdots$

$. : \mathsf{o}$

$. : \mathsf{N}$

$. : \mathsf{P}$

And then our proof objects can be turned into HOL-terms of type S.

Discussion

- We can encode our proof objects (i.e., the semantical objects of interest in MathDIALOG) as terms of some logic L.
- L probably needs to be more expressive than FOL.
- If do such an encoding, what can the ’standard underspecification approach’ of Chorus offer to us?
  Would it be adaptable to more expressive logics than FOL?
  Would the inference support we would obtain be stronger than our self-baken ones?