Integrated Proof Transformation Services

Jürgen Zimmer*

Universität des Saarlandes, Germany

University of Edinburgh, Scotland

joint work with Andreas Meier, Geoff Sutcliffe, Yuan Zhang

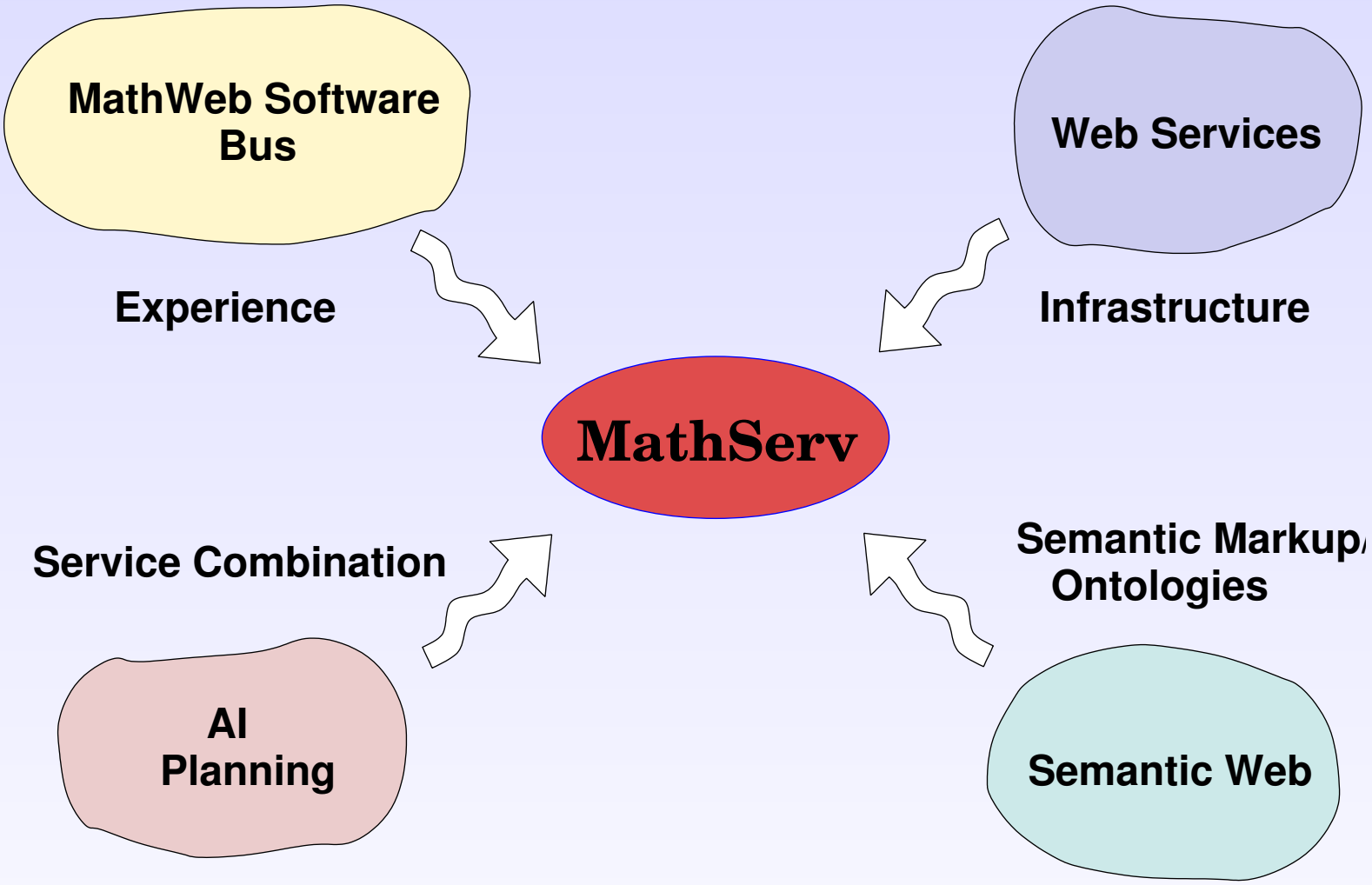
***The author is supported by the European Union**

CALCULEMUS **IHP Training Network HPRN-CT-2000-00102**

Overview

- MATHSERV: Semantic Reasoning Web Services
- Some Systems Integrated in MATHSERV
- Theorem Proving and Proof Transformation Services
- Brokering of Proof Transformation Services
- Conclusion
- Future Work
- $1/2$ Demo

The MathServ Framework



The MathServ Framework

A new framework for semantic reasoning services:

- Based on Web Service technology.
- Semantic markup for web services in the Mathematical Service Description Language (MSDL):
  - $\Rightarrow$ Developed by MONET and MathBroker project.
  - $\Rightarrow$ Based on commonly agreed ontology.
- Brokering mechanism retrieves and combines reasoning services using modified POP planner.

Benefits of MathServ

The MathServ framework can be used by humans or machines to...

- retrieve reasoning services (by human $\vee$ machine) given a semantic description of a problem.
- automatically combine services to tackle a problem.
- tackle subproblems in automatic or interactive theorem proving.

No need to know the underlying reasoning system!

Systems Integrated as Web Services

1) Automated theorem proving systems:

- EP, Otter, SPASS.
- For classical first-order predicate logic with equality.

2) Otterfier Tool for proof transformation [Sutcliffe'04]:

- CNF refutation $\longmapsto$ CNF refutation (BrFP) calculus
  BrFP = Binary resolution, Factoring, Paramodulation
- Calls Otter to replace "alien" inference steps

More Systems Integrated

3) The Tramp system [Meier'00]:

- FOF problem + CNF refutation (BrFP) $\longmapsto$ ND proof at assertion level.

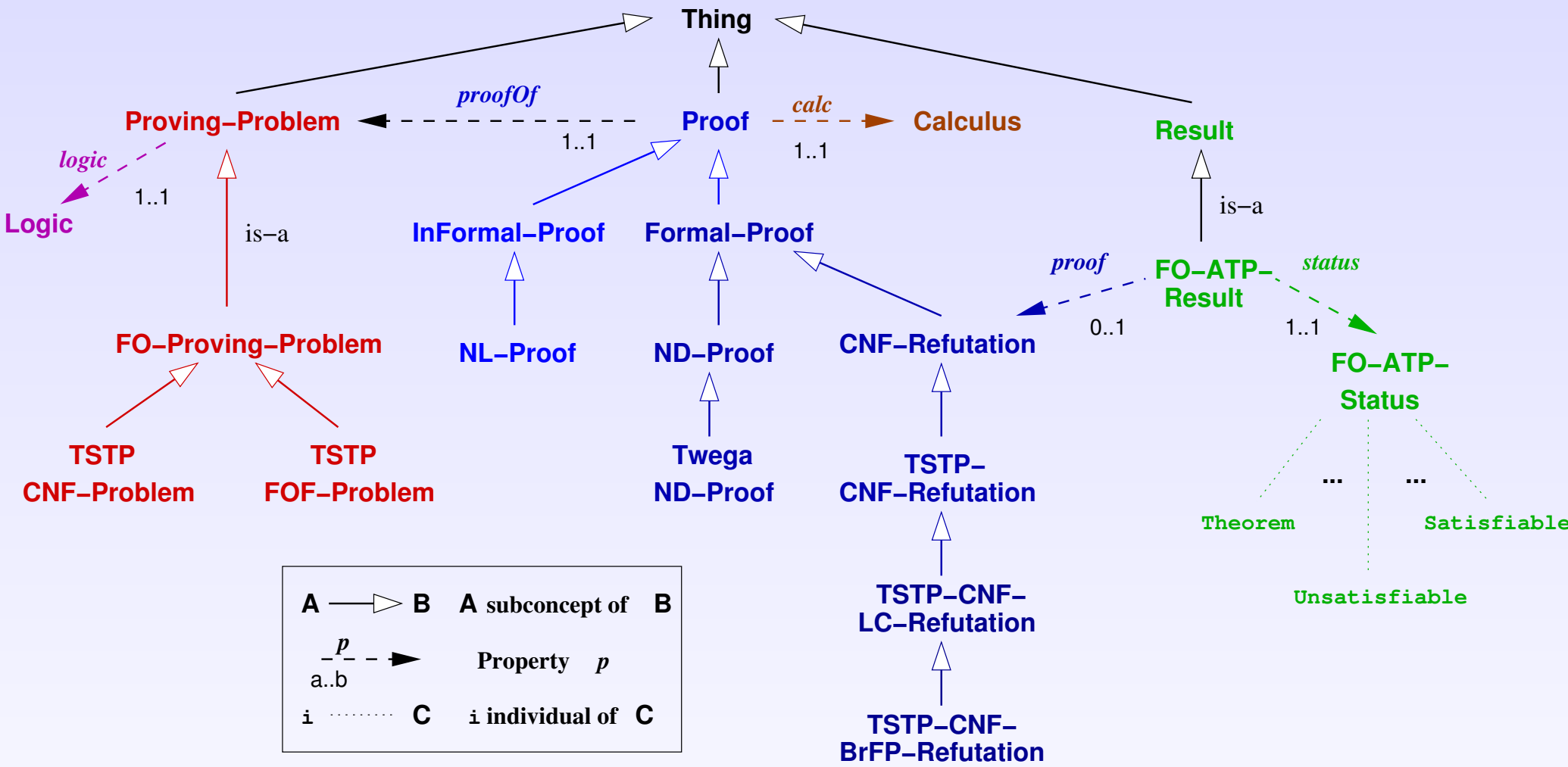
  assertion level step [Huang'94]: $\dfrac{F \subset G \qquad c \in F}{c \in G} \subset\text{DEF}$

  $(\subset\text{DEF}) : \forall s_1.\forall s_2.(s_1 \subset s_2 \Leftrightarrow \forall x.(x \in s_1 \Rightarrow x \in s_2))$

4) The *P.rex* system [Fiedler'01]:

- Formal proof $\longmapsto$ Natural Language (NL) proof.
- Proof quality depends on linguistic knowledge.

An Ontology for Service Descriptions



An ATP Service in MSDL

The central part of an MSDL description [MICAI'04]:

Service: EpATP	
input parameters:	*problem*::TSTP-CNF-Problem (Concept)
output parameters:	*result*::FO-ATP-Result
pre-conditions:	$\top$
post-conditions:	*proof(?result, ?proof)* $\Rightarrow$ *type(?proof, TSTP-CNF-Refutation)*

- We completely omit XML details.
- Conditions in Semantic Web Rule Language (SWRL) (RDF-triples, conjunction, implications).

The Otterfier Service in MSDL

Service: OtterfierService	
input parameters:	*oldResult*::FO-ATP-Result
output parameters:	*newResult*::FO-ATP-Result
pre-conditions:	*proof*(*?oldResult*, *?oldProof*)
post-conditions:	*proof*(*?newResult*, *?newProof*) $\wedge$ *type*(*?newProof, TSTP-CNF-BrFP-Refutation*) $\wedge$ *altProof*(?newProof, ?oldProof)

- *altProof* = alternative proof

The Services of Tramp and P.rex

Service: NDforFOF	
input parameters:	*fofProblem*::TSTP-FOF-Problem *atpResult*::FO-ATP-Result
output parameters:	*ndProof*::Twega-ND-Proof
pre-conditions:	*proof(atpResult, ?proof)* $\wedge$ *type(?proof, TSTP-CNF-BrFP-Refutation)*
post-conditions:	*proofOf(ndProof, fofProblem)*

Service: PrexND2NL	
input parameters:	*ndProof*::Twega-ND-Proof
output parameters:	*nlProof*::NL-Proof
pre-conditions:	$\top$
post-conditions:	*proofOf(?ndProof, ?p)* $\wedge$ *informalProofOf(?nlProof, ?p)*

Example Conjecture

Scenario: Given a first-order conjecture:

- hypotheses: F is a group, $U \subset F$,
  $$\forall x, y.(x \in U \wedge y \in U) \Rightarrow (x \circ y^{-1} \in U) \qquad \text{(Criterion)}$$
- conclusion: $\forall v.v \in U \Rightarrow v^{-1} \in U$
- and some theory axioms, e.g.:
  $$\forall s_1, s_2.\ (s_1 \subset s_2 \Leftrightarrow \forall x.(x \in s_1 \Rightarrow x \in s_2)) \qquad (\subset\text{DEF})$$

Queries:

Peter: Give me a first-order ATP system result!

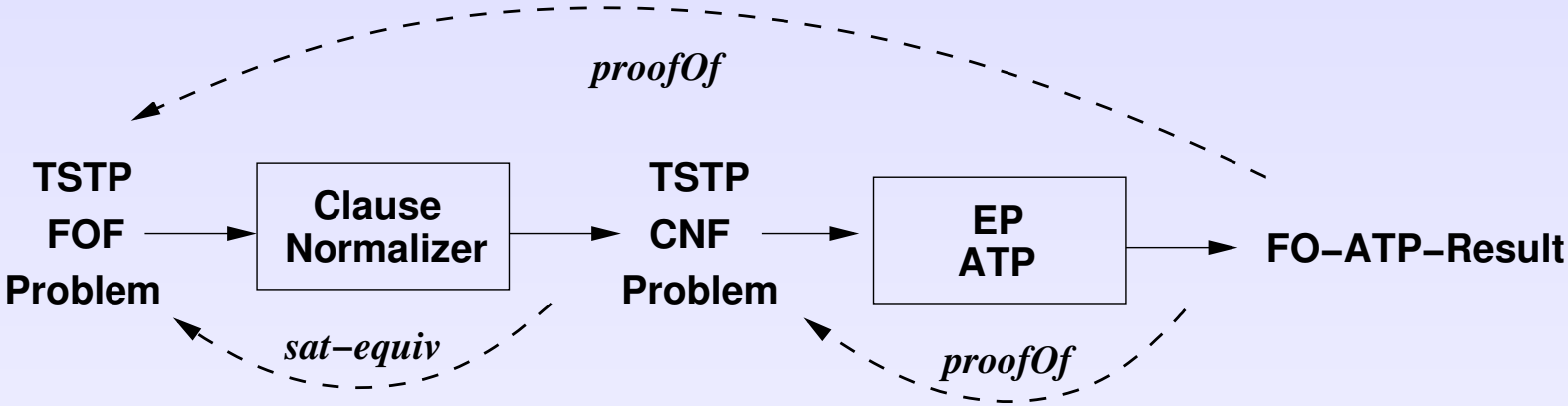
Susan: Give me a ND calculus proof!

Mary: Give me a NL proof!

Example: TSTP Encoding

```
fof(f_group,hypothesis, ( group(f) )).
fof(u_subset,hypothesis, ( subset(u,f) )).
fof(subgroupcriterion,hypothesis, ( ! [X,Y] :
          ( (member(X,u) & member(Y,u))
            => member (multiply(f,X,inverse(f,Y)),u) ))).
fof(subset,axiom, ( ! [S,T] :
         ( subset(S,T)
           <=> ! [X] : ( member(X,S) => member(X,T) )))).
fof(prove_this,conjecture, ( ! [V] :
          ( member(V,u)
            => member(inverse(f,V),u) ))).
```

Peter's Query: Execution Plan

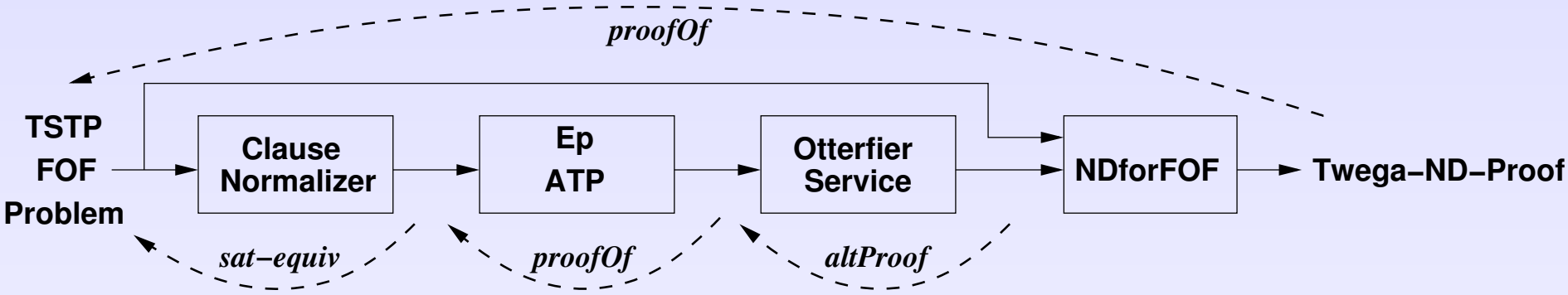


Peter's Query: A Resolution Proof

EP delivers a proof with 19 clauses in 31ms.

```
          ...
cnf(10,axiom,( equal(multiply(X1,X2,inverse(X1,X2)),identity(X1))
             | ~group(X1) | ~member(X2,X1))).
cnf(12,axiom,(group(f))).
cnf(13,axiom,(subset(u,f))).
cnf(14,axiom,( member(multiply(f,X1,inverse(f,X2)),u)
             | ~member(X1,u) | ~member(X2,u))).
cnf(15,conjecture,(member(sk2,u))).
          ...
 cnf(273,derived,(~member(sk2,f)),
    inference(rw,[status(thm)],[270,15,theory(equality)])).
 cnf(274,derived,(false),
    inference(rw,[status(thm)],[273,51,theory(equality)])).
```

Execution plan for Susan's Query

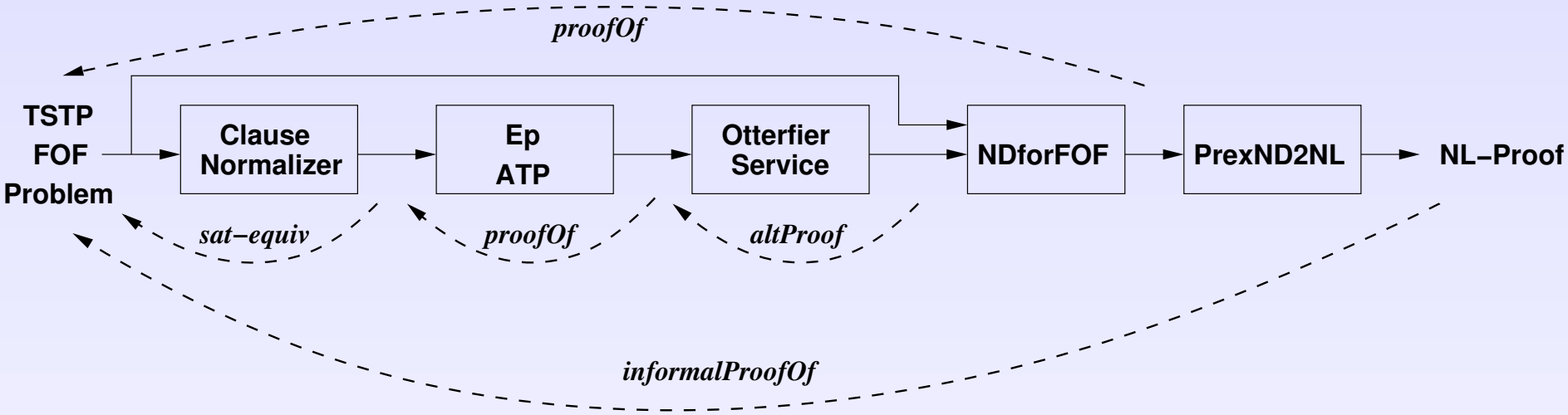


ND Proof answers Susan's Query

Tramp's ND proof with 10 steps (6 assertion level) after 20 seconds:

$L2.$	$L2$	$\vdash member(C, U)$	(Hyp)
$L4.$	$\mathcal{H}_1$	$\vdash member(multiply(F, C, inverse(F, C)), U)$	$(Criterion\ \ L2)$
$L5.$	$\mathcal{H}_2$	$\vdash member(C, F)$	$(\subset DEF\ \ U \subset\ \ L2)$
$L6.$	$\mathcal{H}_3$	$\vdash multiply(F, C, inverse(F, C)) = identity(F)$	$(InvAx\ \ FGroup\ \ L5)$
$L7.$	$\mathcal{H}_4$	$\vdash member(identity(F), U)$	$(=Subst\ \ L4\ \ L6)$
$L8.$	$\mathcal{H}_4$	$\vdash member(multiply(F, identity(F), inverse(F, C)), U)$	$(Criterion\ \ L7\ \ L2)$
$L9.$	$\mathcal{H}_3$	$\vdash member(inverse(F, C), F)$	$(InvAx\ \ FGroup\ \ L5)$
$L10.$	$\mathcal{H}_5$	$\vdash multiply(F, identity(F), inverse(F, C)) = inverse(F, C)$	$(UnitAx\ \ FGroup\ \ L9)$
$L3.$	$\mathcal{H}, L2$	$\vdash member(inverse(F, C), U)$	$(=Subst\ \ L8\ \ L10)$
$L1.$	$\mathcal{H}$	$\vdash member(C, U) \Rightarrow member(inverse(F, C), U)$	$(\Rightarrow I\ \ L3)$
$Conj.\mathcal{H}$		$\vdash \forall x_{\blacksquare} member(x, U) \Rightarrow member(inverse(F, x), U)$	$(\forall I\ \ L1)$

Mary's Query: Execution Plan with P.rex



Mary's Query: NL Proof

P.rex' NL proof after 80 seconds with BASIC linguistic knowledge:

[...] Let $member(C, U)$. Then $member(C, F)$ because $subset(U, F)$ by $\subset DEF$. Thus $member(inverse(F, C), F)$ because $group(F)$ by $InvAx$. That implies that $multiply(F, identity(F), inverse(F, C)) = inverse(F, C)$ by $UnitAx$ since $group(F)$. That implies that $member(multiply(F, C, inverse(F, C)), U)$ by $Criterion$. That leads to $multiply(F, C, inverse(F, C)) = identity(F)$ by $InvAx$ because $group(F)$. That implies that $member(identity(F), U)$. Therefore $member(multiply(F, identity(F), inverse(F, C)), U)$ by $Criterion$. That implies that $member(inverse(F, C), U)$. Therefore $member(C, U)$ implies that $member(inverse(F, C), U)$. That implies that $member(x, U)$ implies that $member(inverse(F, x), U)$ for all x.

Conclusion

The MathServ framework offers

- Semantic retrieval of reasoning services.
- Our broker can provide customized execution plans for a given query.

In Computer-supported theory development:

- Interactive theorem proving useful for
  - closing subgoals.
  - retrieve useful lemmas (HELM web services).
- Proofs in different calculi.

Ongoing and Future Work

- Service Execution
  $\Rightarrow$ MONET plan executor?
- Description of other reasoning systems
  (e.g., model generators, decision procedures?).
- More fine-grained services (like MONET).
  (e.g., given $n \in \mathbb{N}$, prove that n is prime).
- Advanced brokering with
  - reasoning on ontology (subsumption test, etc.).
  - disjunctive plans (or re-planning).