Logosphere
A Digital Library of Formal Proof

Carsten Schürmann

IT University of Copenhagen
Processor Verification

- **INTEL (HOL/HOL light).** [John Harrison]
- $500$mio Pentium bug.
- **AMD (ACL2, Nqthm).** [Matt Kaufmann]
- Siemens, Microsoft (ASM). [Yuri Gurevich]
What’s Intel up to?

**HOL**

[**Harrison’03**]

Floating Point arithmetic.

Round the same way if

\[ |\sqrt{a} - S^*| < |\sqrt{a} - m| \]

\[ |- (\text{precision fmt} = 0) \land \]

\( (\forall m. \ m \ \text{IN midpoints fmt} \)

\[ \rightarrow \ \text{abs}(x - y) < \text{abs}(x - m) \]

\[ \rightarrow (\text{round fmt Nearest x = round fmt Nearest y}) \]
NASA Space Shuttle

Huge PVS libraries developed at NASA.

Algebra, Real analysis, Complex numbers, Directed graphs, Graph theory, Integer division, Abstract orders, Lattices, Fixed Points, Power sets, Trigonometry, Series, Taylor’s theorem etc.

Sharing ok, but how?
Mathematics

Four-Color theorem [Appel, Haken 1976]

Kepler’s Conjecture 2D [Thue 1890]
3D [Hales 1989]
MIZAR [Trybulec’72]

- Reconstruct mathematical vernacular.
- Proof verifier.
- Large body of mathematical knowledge.
- No explicit proof objects.
- Journal of formalized mathematics.
  - On the Hausdorff distance between compact subsets. [Adam Grabowski]
  - Chains on a grating in Euclidean space. [Freek Wiedijk]
Logic Diversification

First-order logic
- Propositional logic
- Higher-order logic
  - Martin Löf's type theory
  - Calculus of Constructions
- Temporal logic
  - Propositional logic
  - Modal logic
- Higher-order logic
- First-order logic
- Vampire
- Nqthm
- Otter
- ACL2
- PVS
- SMV
- Chaff
- Nuprl
- COQ
- LCF
- HOL light
- HOL
- Automath
- AGDA
- OMEGA
- MIZAR
- HOL light
- HOL
- Calculus of Constructions
- Coq
Digital Libraries

FDL library. [Constable 2000]

- Storage, retrieval of mathematical facts.
- Logic dependent.

Logosphere. [Schürmann 2002]

- Logical framework.
- Foundationally uncommitted.
- Theory morphisms.
- Currently under development.
What shall we store?

Semantic meaning of a theorem!

Formulas alone insufficient.

• Logics vary in proof-theoretic strength.

• Example: First-order logic vs. impredicative type theory.

• Semantics-preserving transformations.

\[ \models L_1 F_1 \implies \models L_2 F_2 \]
Meaning of theorems ...

... are mathematical entities expressed as

- Denotations (Domain theory).
- Objects (Category theory).
- \{0, 1\} (Model theory).
- Strategies (Game theory).
- **Syntactic Proofs (Proof Theory).**

Large proofs but small trustworthy checkers.
Rest of this talk

joint work with Mark-Oliver Stehr

- The logic HOL.
- Logical framework LF.
- Nuprl type theory.
- HOL - Nuprl connection.
- Open questions.
HOL

• Higher-order logic [Church ‘40]
• HOL theorem prover [Gordon ‘85]
• Flavor: Isabelle/HOL [Paulson, Gordon ‘92]

Terms: \( e_1, e_2 ::= x \mid \lor \mid e_1 e_2 \mid \lambda x : \tau. e \)
Types: \( \tau ::= o \mid \tau_1 \rightarrow \tau_2 \)
HOL (Typing)

Judgments: $e : \tau$

Rules:

- **imp**
  
  $\top : o \to o \to o$

- **eq**
  
  $= : \tau \to \tau \to o$

  $\vdash u$

  $x : \tau_1$

  $\vdash \vdots$

- **app**
  
  $e_1 : \tau_2 \to \tau_1$

  $e_2 : \tau_2$

  $e_1 \ e_2 : \tau_1$

- **lam$^u$**
  
  $\lambda x : \tau_1.e : \tau_1 \to \tau_2$

  $e : \tau_2$
HOL (Proofs)

Judgments: \( \vdash P \)

Rules:

\[
\begin{align*}
\vdash P & \quad \vdash P \supset Q \quad \text{mp} \quad \vdash Q \\
\vdash Q & \quad \vdash P \supset Q \quad \text{disch} \\
\vdash P = P & \quad \vdash (\lambda x : \tau . P)Q = [Q/x]P
\end{align*}
\]
HOL (Booleans)

\begin{align*}
\text{bool} & \triangleq o \\
\text{true} & \triangleq \lambda x : \text{bool}. x = \lambda x : \text{bool}. x \\
\text{all } P & \triangleq P = \lambda x : \tau. \text{true} \\
\text{false} & \triangleq \text{all } (\lambda x : \text{bool}. x) \\
\text{neg } P & \triangleq P \supset \text{false} \\
\text{P and } Q & \triangleq \text{all } (\lambda R : \text{bool}. (P \supset Q \supset R) \supset R) \\
\text{the } P & \quad \text{(newly declared)} \\
\text{ex } P & \triangleq P (\text{the } P)
\end{align*}
Twelf

- Logical framework LF. [Harper '93]
- Meta-language for deductive systems.
- Judgments-as-types, derivations-as-objects.
- Representation methodology.
  - Higher-order abstract syntax.
  - Captures variable binding.
Twelf (cont’d)

Representing numbers in BS (binary strings).

\[ \⌜79\⌝ = *, 1, 0, 0, 1, 1, 1, 1 \]

Representing judgments in LF.

\[ \vdash P : \text{type} = \vdash \⌜P⌝ \]

Representing derivations in LF.

\[
\begin{align*}
\vdash & P \supset Q \\
\vdash & P \\
\vdash & Q \\
\vdash & \vdash Q \\
\end{align*}
\]

Proofs “R”Us –p.30/49
Twelf’s Strength

Adequacy Theorem: Every HOL derivation $\mathcal{D}$ of $P_1, \ldots, P_n \vdash Q$ can be represented in LF as a canonical object $\llbracket \mathcal{D} \rrbracket : \llbracket Q \rrbracket$ in context $u_1 : \vdash \llbracket P_1 \rrbracket, \ldots, u_n : \vdash \llbracket P_n \rrbracket$. 

![Diagram showing the relationship between HOL terms, Types, Typing, and Derivability, and Logical Framework LF, representing Canonical objects.]

Proofs “R”Us – p.27/49
Twelf Encoding of HOL

tp : type.           %name tp (A B).
--> : tp -> tp -> tp.  %infix right 10 -->.
o  : tp.

tm : tp -> type.           %name tm (H G) (x y P Q R).
=> : tm (o --> o --> o).
== : tm (A --> A --> o).
@ : tm (A --> B) -> tm A -> tm B.  %infix left 15 @.
\ (tm A --> tm B) : tm (A --> B).

|-    : tm o -> type.       %prefix 10 |- .  %name |- D u.
mp    : |- H -> |- H == H -> |- G.
disch : (|- H -> |- G) -> |- H == H.
refl  : |- H == H.
beta  : |- (\ H) @ G == (H G).
sub   : {G:tm A -> tm o} |- H1 == H2 -> |- G H1 -> |- G H2.
abs   : |- \ H == G    <- (\ x |- H x == G x).

bool = o.
true : tm bool = (\ [x:tm bool] x) == (\ [x:tm bool] x).
all : tm ((A --> bool) --> bool)
    = (\ [P:tm (A --> bool)] P == (\ [x] true).
all| = [P] all| @ P.
false : tm bool = all (\ [P] P).
neg  : tm (bool --> bool) = (\ [P:tm bool] P == false.
\\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == Q == R) == R).
\\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == R) == (Q == R) == R).
\|\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == R) == (Q == R) == R).
\|\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == R) == (Q == R) == R).
\\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == R) == (Q == R) == R).
\|\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == R) == (Q == R) == R).
\|\ : tm (bool --> bool --> bool)
    = (\ [P:tm bool] \ [Q:tm bool]
        \all \ [R:tm bool] (P == R) == (Q == R) == R).

the| : tm ((A --> bool) --> A).
the  = [P] the| @ P.
ex   : tm ((A --> bool) --> bool)
    = (\ [P:tm (A --> bool)] P @ (the (\ [x] P @ x))).
ex   = [P] ex| @ P.
Nuprl

- Polymorphic extensional type theory.  
  [Constable ‘86]

- Judgments establishes equality among terms.

- A type is *true* iff it is *inhabited*.

- Many applications.
  - Ensemble (TCP/IP stack).  
    [Kreitz ‘04]
  - Protocol Verification.  
    [Felty et al ‘98]
Translation

- Original idea. [Howe ‘98]
- Syntactic argument. [Meseguer, Stehr ‘01]
- Implemented in Nuprl, replay of proof scripts. [Naumov ‘01]
- Formalized and executable specification. [Schürmann, Stehr ‘05]
Translation (cont’d)

• Booleanas.  
  \[ \text{boolean} = \text{unit} + \text{unit} \]
  \[ \text{tt} = \text{inl bullet} \]
  \[ \text{ff} = \text{inr bullet} \]
  \[ \text{if } e \ e_1 \ e_2 = \text{decide } e (\lambda z. e_1) (\lambda z. e_2) \]

• Proposition-as-types. 
  \[ \text{BOOLEAN} = U_1 \]
  \[ \text{TRUE} = \text{unit} \]
  \[ \text{FALSE} = \text{void} \]
  \[ \text{ALL} = \Pi \]
  \[ \text{=} \text{n=} = \Pi \]
Howe’s Observation

• Axiom of the excluded middle.

\[ \vdash \text{inh} \# \Pi x : \text{BOOLEAN}. x + (x \rightarrow \text{void}) \]

• Lift Booleans to propositions.

\[ \uparrow (e) = \text{if } e \text{ TRUE FALSE.} \]

• Lower propositions to Booleans.

\[ \downarrow (P) = \text{decide (inh P) } (\lambda x. \text{tt}) (\lambda y. \text{ff}). \]

• All important laws verifiable within Nuprl.
Translations-as-Relations

• Relations in Twelf.
  \[
  \text{trans-tp} : \quad \text{tp} \rightarrow \text{nuprlterm} \rightarrow \text{type} \\
  \text{trans-tm} : \quad \text{tm } A \rightarrow \text{nuprlterm} \rightarrow \text{type} \\
  \text{trans-sentence} : \quad \text{tm } o \rightarrow \text{nuprlterm} \rightarrow \text{type} \\
  \text{trans-proof} : \quad \vdash P \rightarrow \text{trans-sentence } P \ T \rightarrow \vdash M \#T \rightarrow \text{type}
  \]

• Defining declarations omitted.

• Executable within Twelf.

• We can transform HOL proofs into Nuprl.
Conclusion

• There is a true need to share mathematical knowledge in form of proofs.
• Proof-theory: syntax instead semantics.
• Logical framework technology important.
• Proof conversion between HOL and Nuprl.
• For other systems (PVS), work in progress.
Open Questions

• Design of a query language.
• Design of the database.
• Shared domains, integers, natural numbers, complex numbers.
• Partial transformations.
• Connection to OMDOC. [Kohlhase 2001]
• Formalization of other logics.
Thank you!

Logosphere.
A Formal Digital Library.

Department of Computer Science
Yale University
51 Prospect St.
New Haven, CT 06520-8285
U.S.A.

Home  Publications  Logics  Resolution  Paramodulation  HOL  Nuprl  PVS  Examples

Retrieval (not yet implemented.)

Logosphere Search  I'm Feeling Lucky.

Stay tuned.
Submission. (Actively under construction.)

Choose File  no file selected
Submit Logosphere File

News
The Economist ran an interesting article March 31, 2005 about Proofs and Beauty. Enjoy.

Carsten Schürmann, Mark-Oliver Stehr. An Executable Formalization of the HOL/Nuprl Connection in Twelf. 11th International Conference on Logic for Programming Artificial Intelligence and Reasoning March 14-18th, 2005, Montevideo, Uruguay. The complete Twelf development source code can be found here: hoi-nuprl.tgz.