Isabelle — a generic theorem proving environment

Markus Wenzel
TU München

March 1998

1. Generic Isabelle
2. Isabelle/HOL
3. Current work at Munich

Generic Isabelle

Isabelle (Paulson, Nipkow, since 1986) provides a powerful generic framework for:

- rapid prototyping of deductive systems
- real working environments (object-logics ZF, HOL, HOLCF)

Meta Logic $\approx$ Church's Simple Type Theory (λ-terms, simple types, type classes)

How to implement object-logics:

1. declare concrete syntax (mixfixes, translations)
2. declare abstract syntax (higher-order constants)
3. declare inference rules (meta-logical formulas)

$\rightarrow$ *primitive inference based on HO resolution*

4. instantiate generic proof tools (e.g. simplifier, classical tableau prover)
5. code special proof procedures

$\rightarrow$ *semi-automated reasoning*

Isabelle/HOL

Main Isabelle object-logic: HOL

Provides packages for:

- (co-)inductive sets and datatypes
- general well-founded recursive functions (TFL)
- lazy data structures (HOLCF)
- extensible records with single inheritance (HOOL)
- ...

Some major case studies (Munich):

- I/O automata: formal meta theory, protocol verification
- `while`-language: equivalence of semantics, compiler correctness, Hoare logic
- Church-Rosser theorems in untyped λ calculus,
- Algorithm $\mathcal{W}$ for ML polymorphic type inference: correctness and completeness
- ...

Many more applications, at Munich and world-wide.

Current work at Munich

Main Isabelle projects at Munich:

Isabelle/Isar (Markus Wenzel)
Intelligible semi-automated reasoning — readable formal proof documents

Isabelle/Bali (David von Oheimb, Cornelia Pusch)
Java and JVM formalization — type system, semantics, compilers

Isabelle/HOOL (Wolfgang Naraschewski)
Formal verification of OO programs