three wishes

Freek Wiedijk

Radboud University Nijmegen

future of ITP workshop

University of Cambridge

2009 08 24, 09 : 30

why wish?

why ITP?

- *for mathematics*
  - correctness
  - explicitness
  - mathematical objects in the physical world
- *for programming*
  - correctness $\approx$ no bugs
  - carefree programming
  - the *pleasure* of crafting a fully correct program

wishes for mathematics

almost-wish: ITP I can sell to the mathematicians

- strong and user programmable automation (HOL)
- integrated declarative proofs and tactic scripts (Isabelle)
- full classical ZFC style set theory (Mizar)
- partiality taken seriously (PVS) $\left(\frac{1}{0}\right)^2 \geq 0$
- dependent and empty types (Coq)
- small kernel implementing small foundations (Metamath)
- mathematical and programming language identical (ACL2)

almost-wish: DNA for formal math

- type theoretical lambda terms
- traces of HOL derivations
- LF
- de Bruijn's $\Delta\Lambda$, aka $\Lambda\Delta$, aka AUT-SL

$$\mathcal{T} ::= * \mid x \mid (\lambda x : \mathcal{T} . \mathcal{T}) \mid (\mathcal{T}\mathcal{T})$$

  identification of λ and Π, no definitions or let-bindings
  unlabeled graphs with four kinds of nodes and two kinds of edges

- *weaker version of* $\Delta\Lambda$

  no convertibility check
  no difference between definitional equality and 'book equality'

problem:

'the category of groups' is not a set

how to talk about 'large categories' in ZFC style set theory?

('universes' are not a nice solution)

almost-wish: 'very large scale formalization' project

- all of undergraduate mathematics

  will take about 140 man · years

or:

- classification of finite simple groups

or:

- Fermat

almost-wish: formal library infrastructure

- *made by a whole community, but not well integrated*
  - Coq's contribs
  - Isabelle's AFP
  - Mizar's MML
- *beautifully integrated, but made by an isolated genius*
  - John Harrison's HOL Light library
  - Georges Gonthier's Ssreflect library

Nijmegen's MathWiki project just started
1 postdoc + 1 PhD student

'Wikipedia for math' + formalizations + 'Proof General on the Web'
Coq + Isabelle + ...

genie, *my first wish*: better automation

progress in proof assistant technology:

- automation of formalized primary school math = 'arithmetic'
- automation of formalized high school math = 'calculus'
- automation of formalized university math

`HIGH_SCHOOL_STUDENT_TAC`

'computer algebra under hypotheses'

$$x \neq 0 \ \wedge \ \left|\ln|x|\right| > 2 \ \wedge \ \int_0^{|x|} t\,dt \leq 1 \ \Rightarrow \ -\frac{1}{e^2} < x < \frac{1}{e^2}$$

should run in less than a second

should run without any arguments

wishes for programming

almost-wish: self-verified ITP

- **Coq in Coq**

  Bruno Barras

  not about the code of the actual system

- **HOL in HOL**

  John Harrison

  about the code of the actual system, but currently

  – code has been a bit simplified (no definitions/polymorphism)

  – no formal relation between OCaml code and its HOL rendering

  – no proofs about parsing/printing (Randy's complaint)

genie, *my second wish*: system for proving ML correct

miniML++

features beyond Coq:

- exceptions
- state

  (just global `ref` variables is enough)
- non-terminating functions

  (my computer has a `^C` !)
- input/output

  other OS related functions

almost-wish: nice system for proving C correct

philosophical question: what should I imagine 'correctness' of

- LaTeX
- Mozilla

to mean?

from the quotes file:

```
V7 /bin/mail source: 554 lines.
1989 X.400 specs: 2200+ pages.
```

a program and a specification are the same kind of thing?

so what does it mean to prove a specification correct?

genie, *my third wish*! system for proving strict conformance

strictly conforming =

program runs the same on all machines =

no undefined behavior, no unspecified behavior

- no dereferenced `NULL` pointers
- no dereferenced dangling pointers
- no array accesses outside the bounds
- no meaningless casts
- no integer overflow
- no dependence on evaluation order `i = i++;`
- *etcetera*

proving correctness without specification

why wish?

needed?

- **first wish** (automated high school mathematics)

  computer algebra under hypotheses

- **second wish** (ML verification)

  Hoare logic for higher order programs in the presence of side effects

- **third wish** (C strict conformance)

  Hoare logic for proving strict conformance