Proof Assistant

By Pfalzgraf Martin
Proof Assistant === Code Analysis
Motivation

23 May 2016

Japan suffered a massive ATM withdrawal operation: 1.4 billion yen stolen
What the solver needs to know

- What do you think your code does
- What your loops should accomplish (or I guess)
First Order Logic

forall y1, y2 ((y1 = y2) -> ~y1 < y2)

((fun <= boo) || (boo >= fun)) -> ~fun = boo

forall x (x < 5 -> f(x) = 6)

How the programmer tells the proof assistant what they think their code is doing
Behind the scenes:

- **Hoare Logic**
  - Models code flow control
  - (if then else, while loops)

- **Separation Logic**
  - Extends Hoare logic
  - \(\sim((\sim\text{emp} \times \sim\text{emp}) \ast \sim \text{True})\)
● Prove that your program is correct

● What is wrong with your code
Solvers

- SBV
  - SMT library in Haskell

- Zhe Hou’s FOASL
  - Separation Logic
Example

// {-@ forall n, a (sum = n * a) @@ sum = (j * a) @-}           Q.E.D
sum := 0;
j := 0;
while (j /= n) {
    sum := sum + a;
    j := j + 1;
}
What else is out there

Welcome to Coq trunk (Nov. 2008)

User error: No focused proof (No proof-editing in progress).
Extending with other languages

- Only needs a language definition
  - Limited to the types of the SMT library
- Could integrate with other libraries that already parse C and Python