Mondex: Z work

Steve King

University of York

RAL workshop: 16 January 2006

3 areas of Z work (at least)

- original specification and development: Stepney, Cooper, Woo as PRG-126 in 2000
- King: automated proof using ProofPower-Z (July-August 2004
- Woodcock: re-specification and proof in Z/Eves (summer 200

Additionally, some *retrenchment* work has been done in Z (Banac Stepney et al, 2004-5)

Here, we discuss mainly the ProofPowerZ work.

Outline

1. Background & Motivation
2. Progress
3. Lessons learnt
4. Future plans?

Background & Motivation

- Z spec and designs published, in sanitised form, as PRG mor
- 'We choose to do rigorous proofs by hand: our experience is t tools are not yet appropriate for a task of this size' [PRG-126]

Goals (pre-GC6)

Long-term: To mechanise, in ProofPower-Z, the proofs in the pul specification and design, *making as few changes as possible already been published.*

Short-term: (over 2-month study leave at QinetiQ Malvern): to le possible about ProofPower-Z, and to start on the long-term go

Background & Motivation (cont)

Personal motivations:

- antidote to increased admin load at York
- long-term unfulfilled interest in automated theorem proving

Wider motivation:

- *possible* case study for GC6

Progress

By the end of August 2004:

- I had a reasonable understanding of the basic use of ProofPo package, use of tactics, etc) for proving Z conjectures. But mo expertise would be required ...
- I had proved that the 3 abstract operations (*TransferOK*, *Tran* maintained the security properties (*NoValueCreation*, *AllValu*
  - 2.5 pages in PRG-126
  - 15.5 pages of my proof script, including lemmas
- I'd started on the refinement proofs: $A \sqsubseteq B$ (100 pages of PR $B \sqsubseteq C$ (30 pages)

Progress (cont)

Some small but significant changes were made to the published t

- missing domain checks: in the context of

  $$f, f' : X \pfun Y \quad ,$$

  a predicate like $f' x = exp$ needs to have an explicit additiona

  $$x \in dom\ f'$$

  Alternatively, it could be changed to $(x, exp) \in f'$. This chan the proof cannot be completed.

- a schema quantification like $\forall x : X;\ S \bullet pred$ (such as is fo function definitions like *totalAbBalance*) is not easy to deal wi this is rewritten as $\forall x : X;\ s : S \bullet pred$, then proofs becom easy to prove a lemma that the two forms are equivalent.

Progress (cont)

- there is an inconsistency between two of the abstract operatio *AbTransferLostTD* has an expression like $f'x = \mu - exp$, whi *AbTransferOkayTD* expresses a similar constraint as $f'x \in \{$ equivalent, as the set has only one member. [This, like other c caused by the sanitisation for publication process.]
- there are several small typos in the *B* and *C* level specificatio refinement proofs. These are recorded in a sheet available fro Stepney's PRG-126 webpage: recommended if you are readi carefully.

Lessons learnt (in 2004)

- it was easier than I expected to learn ProofPower-Z
  - but documentation on basic use could be improved
- the ‘sanitisation for publication’ process is not easy, and is the oddities:
  - empty schema (caused by hiding all components)
  - *allLogs* : two similarly named components were merged
- for real proof examples, size of screen display is important: do
- mechanical theorem-proving is fun!

Progress since late 2004

Future plans?

In late 2004, my plans were:

- continue work on refinement proofs
  - can the structure of the hand proof be maintained?
  - can it be improved?
- comparison with Jim's work using Z/Eves
- ? automating the proof

Progress has been slow, but ...

Acknowledgements

- Systems Assurance Group, QinetiQ, Malvern.
  - Colin O’Halloran
  - Alf Smith, Mark Adams, Phil Clayton
- Mondex authors, for answering queries

References

- for details of Mondex (& MultOS) publications:
  `http://www-users.cs.york.ac.uk/ susan/`
- for corrections etc to Mondex specs:
  `http://www-users.cs.york.ac.uk/ king/p`

JCPW’s work in Z/Eves

- Aim was to re-express the Mondex specification, in Z, but tailo proof
- Presented in detail to RefineNet workshop on Mondex, Septe

Original state

$AbPurse == [balance, lost : \mathbb{N}]$

$[NAME]$

$AbWorld == [abAuthPurse : NAME ⇻ AbPurse]$

JCPW's state

$[NAME]$

$$
\begin{array}{|l}
\textit{AbWorld} \\
\hline
index : NAME ⤔ \mathbb{N} \\
credit, debit : \mathrm{seq}\,\mathbb{N} \\
balance, lost : NAME ⇸ \mathbb{N} \\
\hline
\mathrm{ran}\ index = \mathrm{dom}\ credit \\
balance = index \fatsemi credit \\
lost = index \fatsemi debit \\
\hline
\end{array}
$$

Proof based around summing sequences, and an *update* function

$$\mid \quad \textit{update} : (\text{seq } \mathbb{Z}) \times \mathbb{Z} \times \mathbb{Z} \rightarrow \text{seq } \mathbb{Z}$$

$$\textit{update}(s, i, n)$$

Express state change as 2-stage update:

$$\textit{mid} = \textit{update}(\textit{credit}, \textit{from}, (\textit{credit}(\textit{from}) - \textit{value}?))$$
$$\textit{credit}' = \textit{update}(\textit{mid}, \textit{to}, (\textit{mid}(\textit{to}) + \textit{value}?))$$

First attempt: develop theory of results about *update*, based on in

Then: re-define *update* axiomatically, based on $\textit{sum}(\textit{update}(s, i,$

Effect on proofs

- domain checks (because of Z/Eves)
- finiteness (because of Z sequences)
- generic theorems (not well supported by Z/Eves)

Final proof

that 3 abstract operations maintain safety properties

10 definitions, 15 theorems, 20 proofs

Proof steps:

prove / prove by reduce / rewrite	22
prenex / simplify	4
cases / next	3
instantiate	3
apply / use	15
	47

Conclusions

- Two days' effort to produce radical recasting of Mondex spec
- Much simpler spec: how would the refinement look, based on
- *Getting the job done* by *exploring the theory*