Mondex with the Alloy model-finding method

Tahina Ramananandro[1] and Daniel Jackson[2]

[1] École Normale Supérieure, 75005 Paris (France)
http://www.eleves.ens.fr/~ramanana/work/mondex
[2] Massachusetts Institute of Technology, Cambridge MA 02139 (USA)
http://alloy.mit.edu

Specifying Mondex with the Alloy specification language. Starting from the existing specification in Z, we wrote a specification of Mondex in the Alloy specification language, based on relational first-order logic with transitive closures.

There is no notion of records, so, contrary to Z, Alloy does not need $NAME$s to disambiguate two objects having the same field values.

We have chosen to represent amounts through sets of coins. So, we modelled a system similar to physical coins instead of integers. We had to develop constraints to prevent purses from sharing common coins in their balances. Then, values can be computed through relational operators gathering the corresponding coins.

But infiniteness cannot be expressed with the Alloy logic, so all those issues have been dropped in our work.

If finiteness issues are dropped, then it is worth noting that the Mondex specification can be expressed in first-order logic even without transitive closures.

Checking the model with Alloy Analyzer, a model finder. We checked the specification with Alloy Analyzer, a piece of software developed by the Software Design group at MIT and based on model-finding : it translates the model into a boolean formula which it tries to satisfy. If an assignment of variables is found, then the program translates it back to get a counterexample.

However, such a translation is possible only for a predefined scope, that is given a finite bound on the number of objects in an instance. The model has been checked for a scope of at most 8 objects per kind (8 coins, 8 purses, . . .) but considering only the relevant pre-states and post-states for each operation.

Counterexamples allowed us to find the following bugs :

- Purses can hold unauthentic transaction details.
- Wrong case analysis in the proof of the Abort/AbIgnore refinement.
- The ΦBOp framing schema does not take the *ether* into account.

Further work. The Alloy model-finding method is useful to find bugs in the specification ; however, it provides no proof. But it could be interesting to get first-order theories from the Alloy model and submit them to automated theorem provers. However, first attempts gave quite few results : only theorems relevant to the Abstract world security properties have been proved.