Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

RESEARCH OUTPUTS / RÉSULTATS DE RECHERCHE

Poster: Behavioral Model-Based Testing of Variability Intensive Systems

Devroey, Xavier

Publication date:
2015

Document Version
Peer reviewed version

Link to publication

Citation for pulished version (HARVARD):
Devroey, X 2015, 'Poster: Behavioral Model-Based Testing of Variability Intensive Systems', Journée des Doctorants en Informatique 2015, Namur, Belgium, 26/07/14 - 26/07/14.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

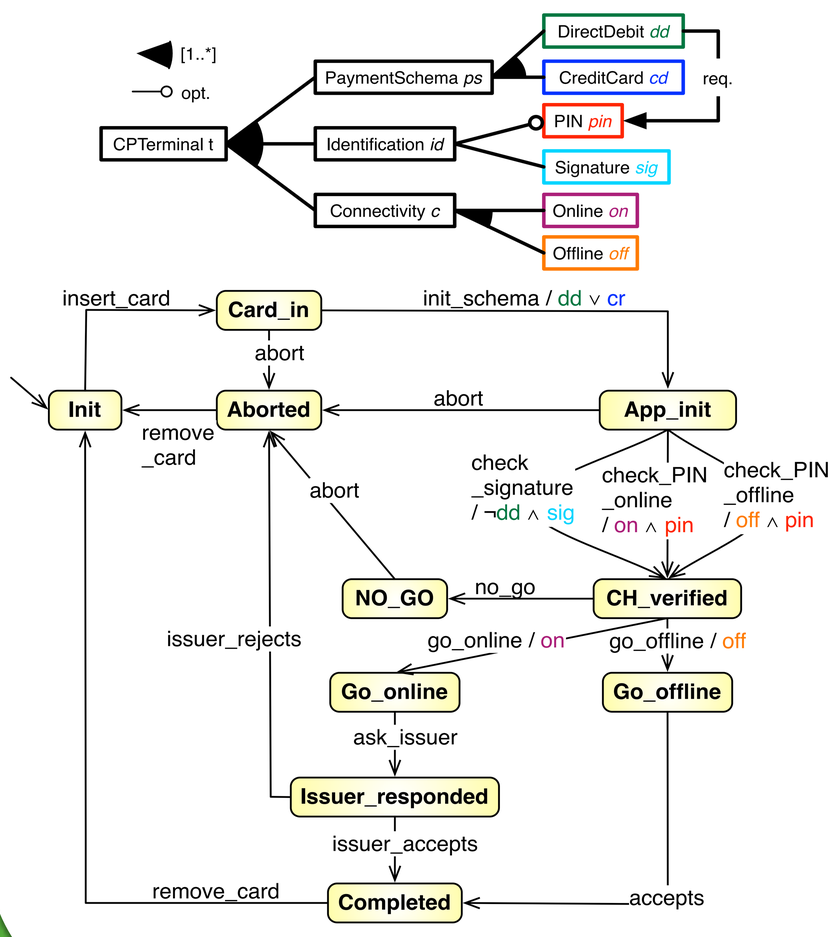
Download date: 14. May. 2024

Behavioral Model-Based Testing of Variability Intensive Systems

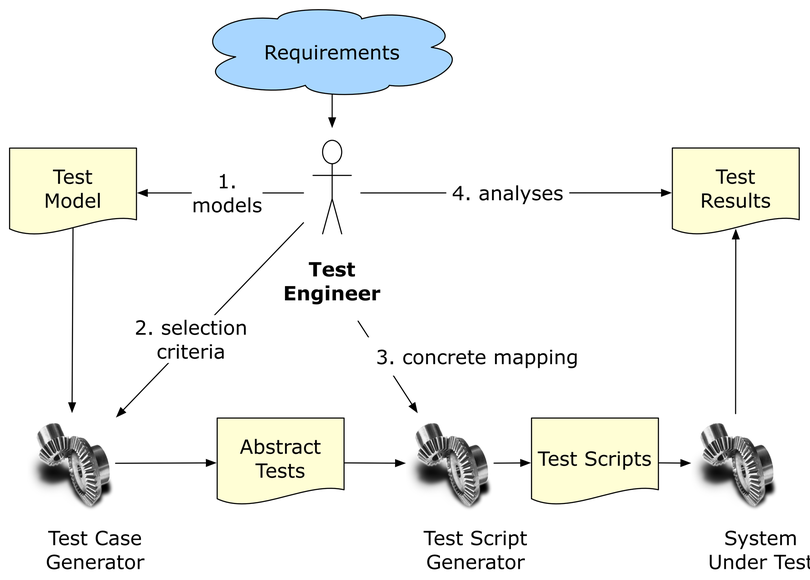
Xavier Devroey, Gilles Perrouin, Pierre-Yves Schobbens, Patrick Heymans

- Focus on **Variability-Intensive Behavioural** Models (software product lines, plugin-based systems, etc.)
- Formal Foundations (**Featured Transition Systems**) suitable for **Analysis** and **Checking**
- Combining **Model-checking** and **Test-case Generation** techniques in **VIBeS Framework**
- Cover various **Testing Activities** (test-case seleciton, prioritization, mutation assessment, etc.)

Test Model

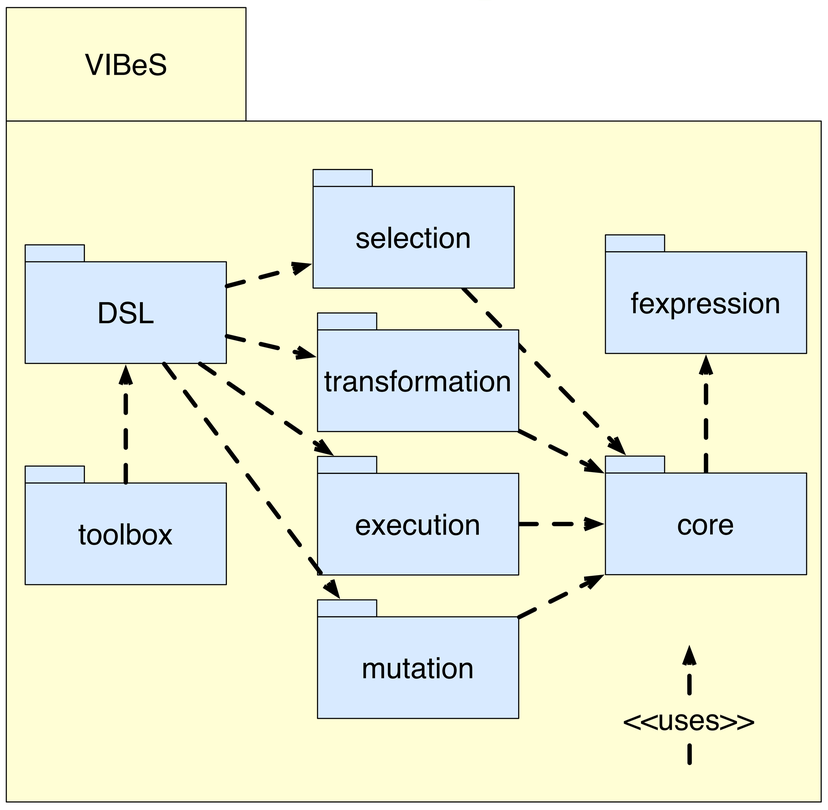


Model-Based Testing Process



1. Models definition (Java DSL)
2. Selection criteria definition
   - Maximizing
     - Structural coverage (state, transition, paths, ...) [2,3]
     - Usage coverage (based on a usage model) [4]
     - Dissimilarity measure (based on distance def.)
     - Number of mutants killed by the tests [1]
   - Minimizing
     - Number of test cases in the test set [3]
     - Number of valid configurations needed by the test set [3]
3. Concrete mapping definition (*TBD*)
4. Tests results analysis (*TBD*)

Variability Intensive system Behavioural teSting Framework



Models Def.

Featured Transition System Java DSL

```
public class CPTerminalFts extends
   FeaturedTransitionSystemDefinition {
@Override
protected void define() {
  initial("i");
  from("i").action("insert_card")
     .to("CardIn");
  from("cardIn").action("init_schema")
     .fexpr("dd || cr").to("App_init");
  ...
}}
```

Feature Model

Boolean Conjunctive Normal Form (CNF) in DIMACS format used as input to SAT and BDD solvers

Selection Criteria Def.

Structural coverage

```
FeaturedTransitionSystem fts =
  new CPTerminalFts().getTransitionSystem();

SolverFacade solver =
  new Sat4JSolverFacade(dimacsFile);

TestSet set = allStatesSelection(fts, solver);
```

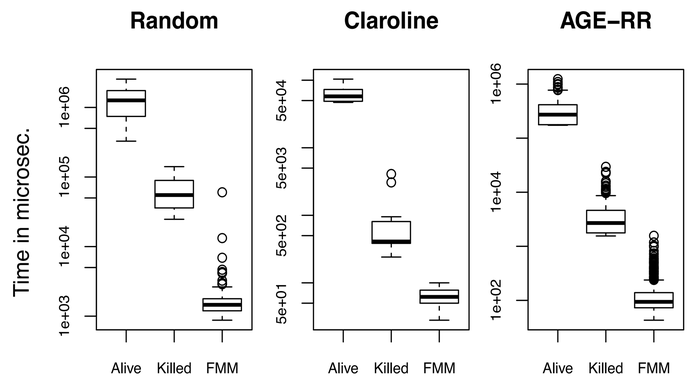
Dissimilarity measure maximization

```
...
set = dissimilarLocalMaximumDistance(fts,
  solver, nbrTestCases, time);
...
set = dissimilarGlobalMaximumDistance(fts,
  solver, nbrTestCases, time);
```

Model Based Mutation Testing

- **Goals**: Quality Assessment (QA)
  - **Tester**: QA of a **set** of abstract test cases
  - **Researcher**: QA of a test case **selection method**
- Mutant = Test Model with injected fault(s)
- Result of the application of a **mutation operator**
  - Act. Exchange, State Missing, Trans. Dest. Exchange, etc.
- **Execution** of the test cases on the mutants
- Mutation score = $\frac{\text{Number of mutants detected}}{\text{Total number of mutants}}$

Mutant Execution using Featured Mutants Model



Future Work

- **Higher order** and **equivalent** mutants detect./gen.
- **Test cases** generation using counter examples in ProVeLines FTS **model checker**
- Abstract test cases **concretization**
- Empirical assessment on an **industrial case study**

Bibliography

[1] Devroey, X., et al. (2014). **A Variability Perspective of Mutation Analysis**. In *Vision & Challenges @ FSE'14*. Hong Kong: ACM.

[2] Devroey, X., et al. (2014). **Abstract Test Case Generation for Behavioural Testing of Software Product Lines**. In *SPLat @ SPLC'15*. Florence, Italy: ACM.

[3] Devroey, X., et al. (2014). **Coverage Criteria for Behavioural Testing of Software Product Lines**. In *ISoLA'14*. Corfu, Greece: Springer.

[4] Devroey, X., et al. (2014). **Towards Statistical Prioritization for Software Product Lines Testing**. In *VaMoS'14*. Nice, France: ACM.

[5] Devroey, X., et al. (2012). **A Vision for Behavioural Model-Driven Validation of Software Product Lines**. In *ISoLA'12*. Crete, Greece: Springer.

Contact:
xavier.devroey@unamur.be

https://projects.info.unamur.be/vibes/

UNIVERSITÉ DE NAMUR

precise

UNIVERSITY OF NAMUR — FOCUS — FOCUS RESEARCH GROUP

Wallonie

fnrs LA LIBERTÉ DE CHERCHER