



A new approach for software testability

Lydie du Bousquet

Laboratoire d'Informatique de Grenoble (LIG)



Test and Testability

- Test is a validation method
- Widely used in Software companies
- BUT software testing is **expensive**
 - Time
 - Funds
- Idea : Design systems **easy** to test
- Notion of **testability**

Software testability

- An estimation of the **testing effort**
- Considering the system (or the process)
- Several definitions
 - "Testability is the **effort** needed for testing"
 - "Testability is the relative ease and expense of **revealing software faults** " (BINDER)
 - "The degree to which a system or component facilitates the **establishment of test criteria** and the performance of tests to determine whether those criteria have been met " (IEEE)



Testability common practices

- Captured with metrics
- Lots of metrics :
 - number of tests to produce (scope metrics)
 - effort/time to produce tests (complexity metrics)
 - observability / controllability
 - probability to discover an error (PIE, DRR,...)
- Metric definitions are related to
 - testing processes,
 - strategies, methods,
 - adequacy criteria
 - Informal feelings ...



Testability :

Let us change the point of view

- Limits of the metrics
 - Difficult to compute, to use, or to interpret !
 - Not validated (theoretically, empirically)
 - Predicting precise testing effort by one (small set) of metrics is elusive
- We need to **improve testability**
 - Collect best practices (specification, design, coding)
 - Build a catalogue of testability pattern
 - Impose the usage of a subset of patterns
 - Evaluate how much those patterns are systematically applied



Example

- Testability pattern for observable classes
 - Each class should have a **reporter** method (R)
- For each classes of the SUT
 - Check if the **reporter** method is present
e.g. 2 classes out 100 have no **reporter**
Application of (R) : **98 %**
 - Check if reporter is correctly implemented



Advantages and issues

- Advantages

- **Flexibility** : patterns chosen upon needs
- Easy to **interpret** and to **use** : modify the places where pattern are not/incorrectly applied
- Adequacy criteria

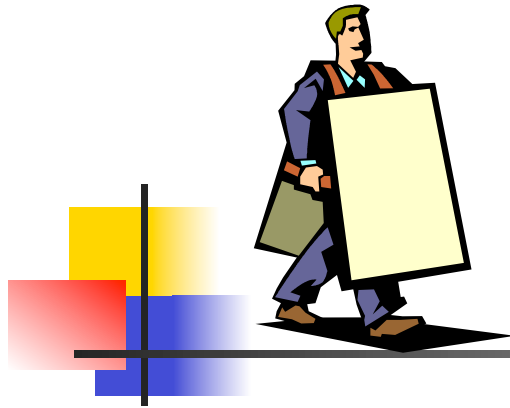
- Issues

- Ability to identify testability patterns ?
- Ability to detect their (good) usage ?



Current work and perspectives

- Currently
 - Collecting testability and anti-testability patterns (Binder's Book, LeTraon, ...)
- To do
 - Validating test patterns (issue)
 - Implementing a environment



Advertisement

- 2 workshops collocated in ICST 2011
- Mutation Testing
- Scenario-based Testing

- Send articles !





A new approach for software testability

Lydie du Bousquet

Laboratoire d'Informatique de Grenoble (LIG)