IC01

On direct guidance of the tester based on model of the tested application

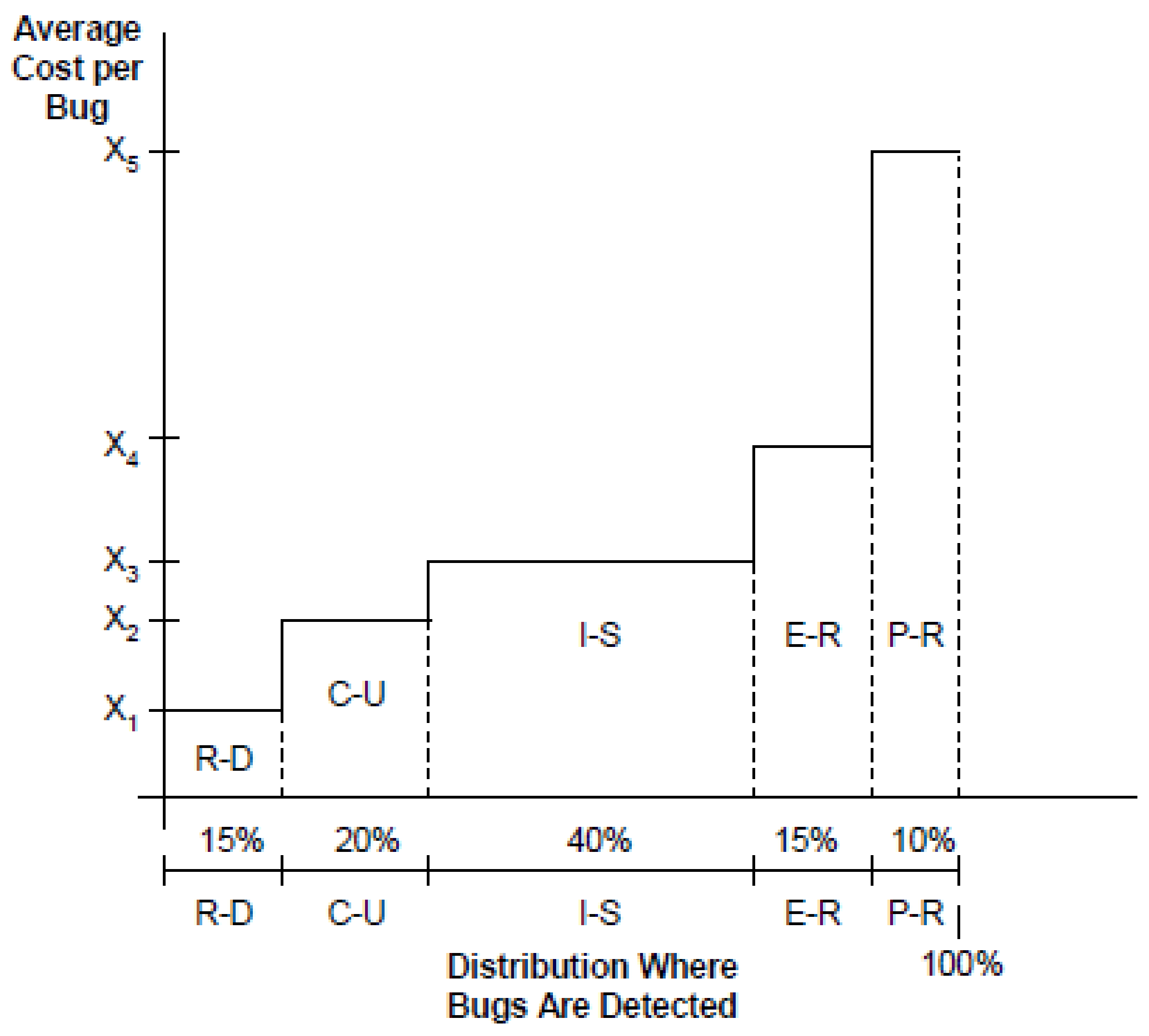
Karel Frajták

Department of Computer Science and Engineering, Faculty of Electrical Engineering
Czech Technical University in Prague

Introduction

The key aspects to successful web application are low error rate, quick error fixing and its reliability. Errors discourage the target users of the application. According to the testing theory it is quite impossible to deliver 100% reliable application. Efficiency of testing is very important, by choosing the right tool the time between discovering the error and its elimination can be reduced. This has positive impact on project economics and quality of the developed application.

In testing stage most errors are discovered and fixed but some errors "leak" into the production stage. Fixing an error in later stages of software development life cycle is expensive.



Keywords

model–based testing, test automation, test case scenario generation, web–based applications, software development life cycle, quality assurance, test coverage

Problem I: Manual Test Case Preparation

Current state:
Test case scenarios are created manually and passed to tester as Word or Excel documents, who then follows the scenario and creates testing report.

- This process is slow, expensive and inaccurate
- Measuring of test coverage is complicated
- Direct relationship between the application and tests does not exist $\rightarrow$ tests are ageing
- What to do when time to create test case scenarios for larger scale applications is running low?

Solution:

- automatic generation of precise and ageless test case scenarios based on the formal model of the system

Advantages:

- achieving high test coverage of tested application through smart test case generation
- achieving very low number of undiscovered errors

Problem II: Verification of Tester's Work

Current state:
Tester follows follows the steps of test case scenario and creates testing report.

- There's no guarantee tester did his/hers task.
  "There was no error at all. I think."
- Given feedback is out of context and vague
  "There was an error when I clicked this."

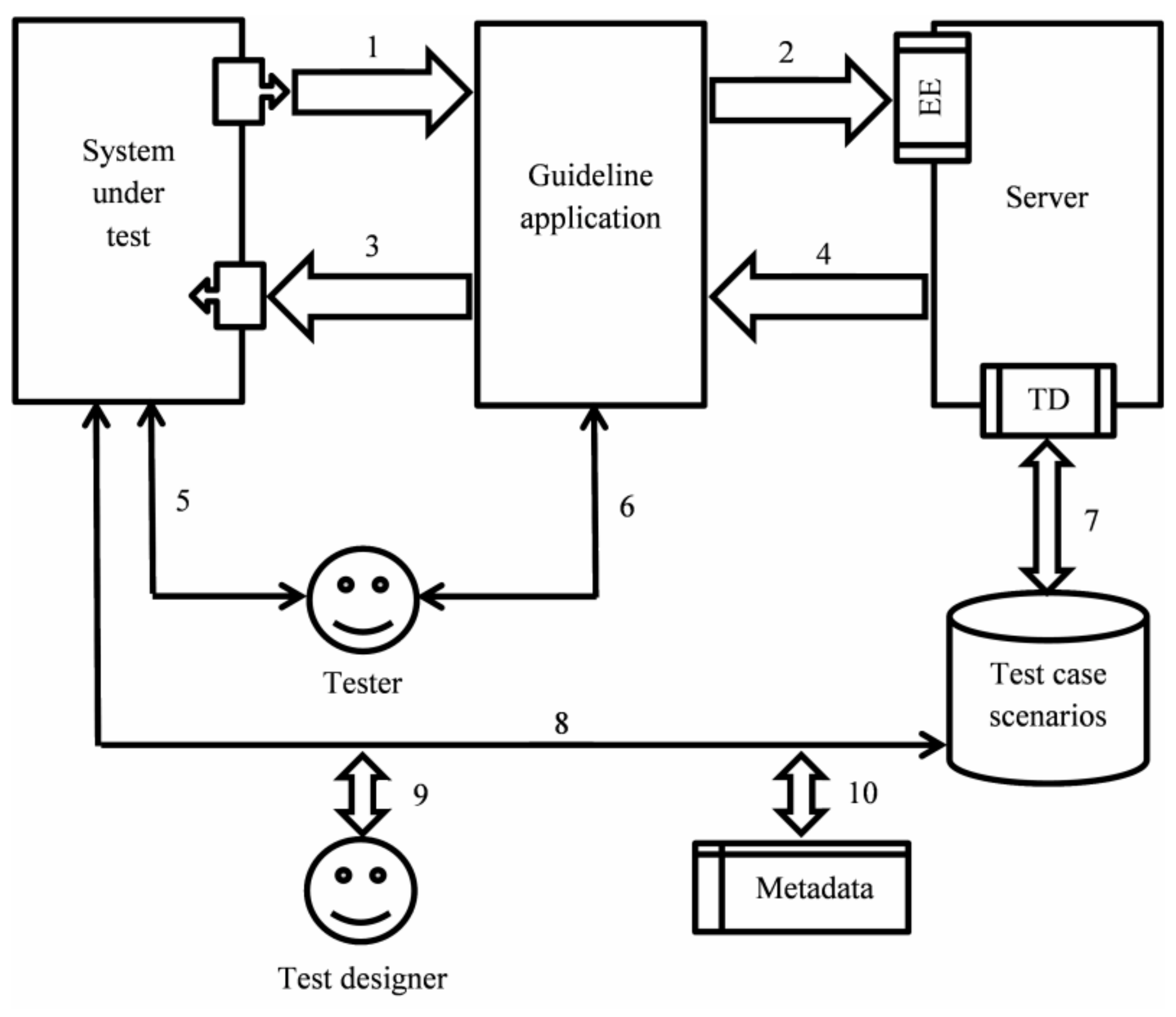
Solution:

- a testing framework based on direct guidance of a tester through testing process
- tested system extensions emulating test case scenario "checkpoints"

Advantages:

- testing process thoroughly verified and monitored
- ability to collect metrics for measuring the quality of tested application and monitoring
- advanced features: smart test dispatch, priority testing, user interaction tracking, interactive help system, etc.

Guideline System Architecture



1. **Tester interaction:** Tester interacts^{5} with SUT and guideline system^{6} at the same time. Test case scenario is loaded from the database^{7} by the test dispatcher^{TD} and passed^{4} to guideline system. Each step in SUT is passed to guideline system through extension points^{1} added to SUT. Constraints for each step are passed to server^{2} and evaluated in expression engine^{EE}.
2. **Test case generation:** Scenarios are generated^{8} or created manually by test designer^{9}. Metadata for non–model requirements are applied to generated test cases^{10}.

Conclusion

Concept of innovative guideline system for improved and verified process of testing of the web application was presented. Pilot implementation is currently in progress. Proposed system will be verified on testing of real world applications.

Acknowledgements

Research was supervised by Ing. M. Bureš, PhD., FEE CTU in Prague and supported by the CTU under grant No. SGS11/157/OHK3/3T/13 and by MSMT under research program No. 6840770014.

Czech Technical University in Prague, Karlovo nám. 13, 121 35 Prague 2, CZ Mail: `frajtak@fel.cvut.cz` WWW: `http://webing.felk.cvut.cz`