

Test prioritization based on failure probability

Michael Shintaku, Matthew Staehely; shintaku, mstaehel

- Testing takes time
- Setting priorities makes better use of developer time
- Is probability-based prioritization effective?



Our Approach

- Build the tool- use simple probability calculation
- Categorize and allow for selection of tests by priority AND/OR by total time
- Calculate time saved by focusing on relevant selected high probability tests

Distinctions

- Focuses on known faults
- Priority strictly probabilistic
- Testing time is the main concern



High

Med

Low

Max Time: xxxxxx

RUN

Test Result Output

- unitTest1(): failure 62%, time 5.682 ms FAILED
- unitTest3(): failure 55%, time 1.63 ms FAILED
- unitTest5(): failure 55%, time .83 ms FAILED

- unitTest2(): failure 25%, time 1500.6 ms FAILED
- unitTest6(): failure 24%, time 2500.0 ms FAILED
- unitTest7(): failure 23%, time 4.36 ms PASSED
- ...
- ...
- unitTestN(): failure 20%, time 1.63 ms PASSED

- unitTest4(): failure 0%, time 4600.0 ms PASSED
- ...
- ...

Tests Run: N Failed: N-F Passed: F
Total Time: too many ms