

Dimensions of Test Coverage

Quantifying What Has and Hasn't Been Tested



RBCS

**TIME TESTED.
TESTING IMPROVED.**

www.RBCS-US.com



Introduction

- What do we mean by “test coverage”?
- What do outside stakeholders hear when we talk about “test coverage”?
- What’s the difference between “test coverage” and “test execution status”?
- Confusion abounds, and leads to misunderstanding of status and bad project management decisions
- Can we build a better understanding of how to measure coverage, along with the strengths and weakness of these measures and how to use tools to gather them?
- Yes!



Test Coverage, Generally

- Test coverage is generally about building confidence
 - In sufficiency and proper execution of testing (proof we tested what we said we would)
 - In proper operation of system (assuming test results are good)
- Building confidence is one of four typical test objectives, which also include
 - Detect defects, especially important defects
 - Reduce risk to an acceptable level
 - Provide information to guide decision-making about project and product
- Metrics can be used to measure the extent to which these objectives are being achieved
- These test coverage metrics should be reported as part of test execution status



Common (and Uncommon) Dimensions

- ⊕ Specification (esp. requirements) coverage
- ⊕ Risk coverage
- ⊕ White-box (esp. code) coverage
- ⊕ Supported configurations (and combinations thereof)
- ⊕ Data set
- ⊕ Personas
- ⊕ Defect taxonomy



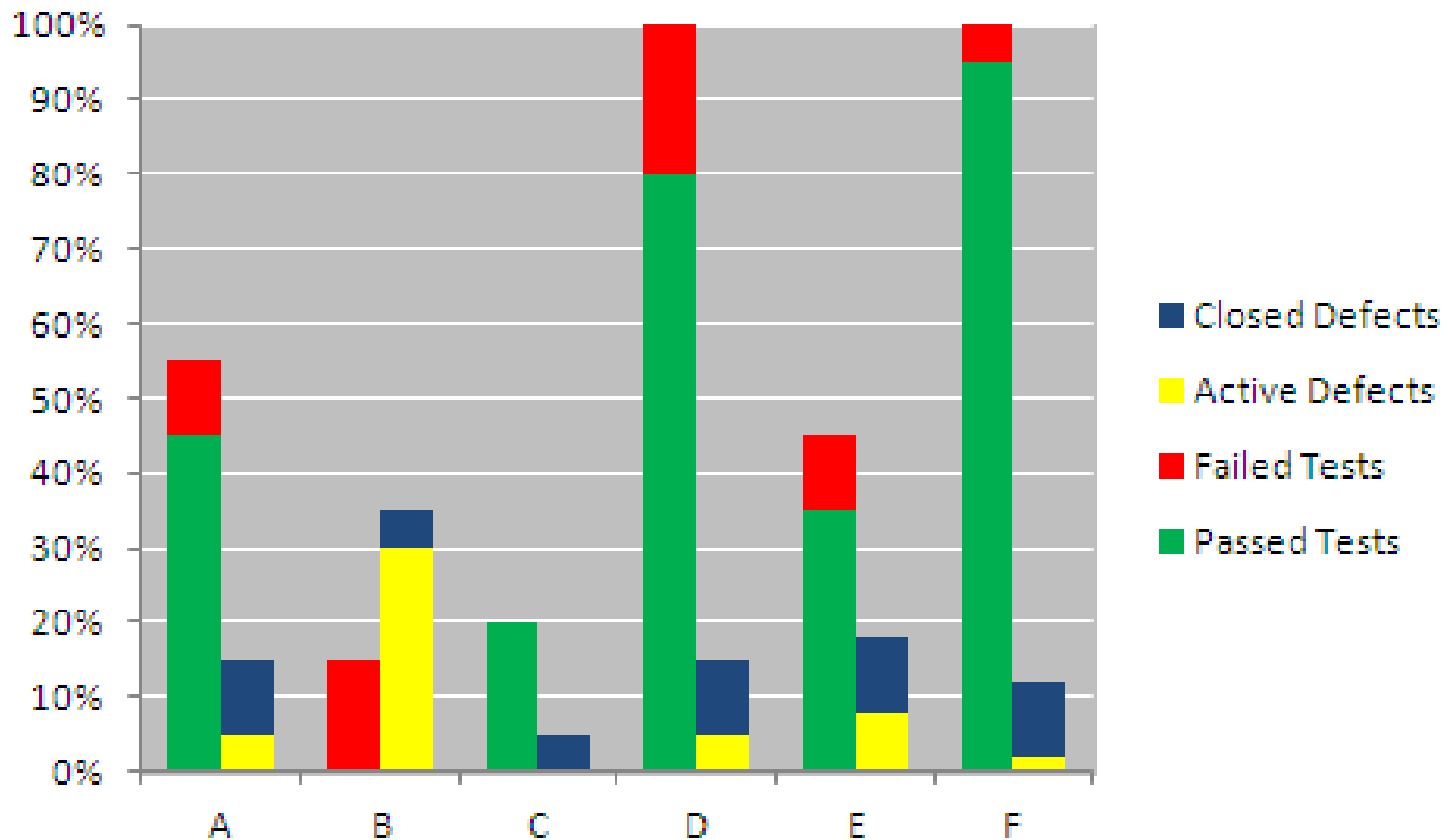
Specification Coverage

- ✚ What it is: The degree to which defined behaviors and/or structures have been tested
- ✚ How's it measured: Percentage of defined requirements, use cases, design elements, screens and fields, or user stories tested, along with test status (pass/fail)
- ✚ Strengths: Flags situations where testing is clearly incomplete or where system elements are weak
- ✚ Weaknesses: Gaps in the specification can create false confidence
- ✚ Tools: Test management and Agile task management



Example of Specification Coverage

Coverage Breakdown



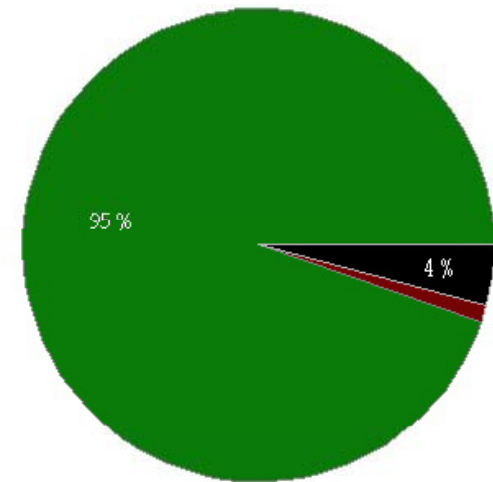
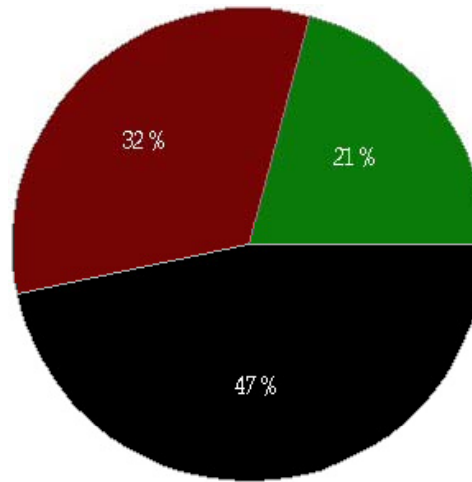
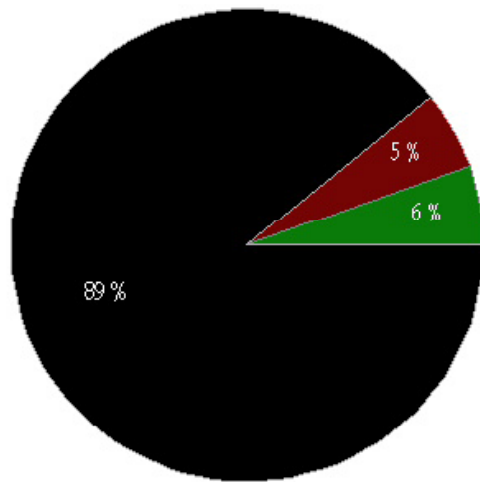


Risk Coverage

- ⊕ What it is: The extent to which identified quality risks have been mitigated
- ⊕ How's it measured: Risk-weighted percentage of identified risks tested, along with test status (pass/fail)
- ⊕ Strengths: Clear measurement of the residual level of quality risk
- ⊕ Weaknesses: Requires good quality risk analysis process
- ⊕ Tools: Test management tools, but customization is usually required



Example of Risk Coverage



Test execution period

Region in green represents risks for which all tests were run and passed and no must-fix bugs were found. Region in red represents risks for which at least one test has failed and at least one must-fix bug is known. Region in black represents other risks, which have no known must-fix bugs but still have tests pending to run.



White-box Coverage

- ✚ What it is: The degree to which the code (and sometimes data) has been tested
- ✚ How's it measured: Varies, depending on the specific metrics (next slide)
- ✚ Strengths: Can reveal under-tested areas of code and help programmers do thorough unit tests
- ✚ Weaknesses: Even very thorough unit testing can only find about 50% of defects, and you can't reveal missing code with white-box coverage
- ✚ Tools: Code instrumentation tools and unit test tools



Examples of White-box Coverage Metrics

- ❖ Statement: % of statements
- ❖ Decision: % of possible outcomes (T/F)
- ❖ Condition: % of atomic conditions (T/F)
- ❖ Modified condition/decision: % of atomic condition that have alone determined outcome twice (T/F)
- ❖ Multiple condition: % of all possible combinations of condition in a decision
- ❖ Basis path: % of McCabe basis paths



Configuration Coverage

- What it is: The extent to which supported configurations have been tested
- How's it measured: Percentage of supported configurations (or configuration pairs) tested, along with test results
- Strengths: Especially for highly portable applications (e.g., mobile apps, browser-based apps), can flag untested hardware and software
- Weaknesses: These metrics sometimes motivate unnecessary pairwise testing, hard to devise a metric that deals with combinations properly
- Tools: Can use test management tools



Data Set Coverage

- ⊕ What it is: Extent to which specific types of data have been tested
- ⊕ How's it measured: Specify the different types of data, then include as part of specification coverage measures
- ⊕ Strengths: Makes visible gaps in testing certain types of data
- ⊕ Weaknesses: Requires a good understanding of the data, which isn't always available in the test team
- ⊕ Tools: As with specification coverage



Persona Coverage

- What it is: Extent to which identified types of users have been included
- How's it measured: Specify the different types of users, then include as part of specification coverage measures
- Strengths: Especially for mass-market software, gets people to pay attention to user diversity
- Weaknesses: Requires a good understanding of the users and their styles of interacting with the application, which isn't always available in the test team
- Tools: As with specification coverage



Defect Taxonomy Coverage

- What it is: Extent to which we've tested for common defects
- How's it measured: Percentage of defects tested, often severity- or frequency-weighted
- Strengths: Helps avoid the "oh no, not again" syndrome after release
- Weaknesses: When production defect information is not available for the current application, can result in over-testing
- Tools: Test management tools, but customization may be required



Conclusions

- Test coverage metrics can give us and other project stakeholders essential insights into our testing
- Unidimensional coverage metrics can be misleading
- Use a balanced combination of coverage dimensions to provide a complete picture to management decision-makers
- Educate people on the coverage metrics, their strengths and weaknesses, before use
- Use tools where possible to automate the process



To Contact RBCS

For over 20 years, RBCS has delivered consulting, outsourcing and training services to clients, helping them with software and hardware testing. Employing the industry's most experienced and recognized consultants, RBCS advises its clients, trains their employees, conducts product testing, builds and improves testing groups, and hires testing staff for hundreds of clients worldwide. Ranging from Fortune 20 companies to start-ups, RBCS clients save time and money through improved product development, decreased tech support calls, improved corporate reputation and more. To learn more about RBCS, visit www.rbc-us.com.

Address: RBCS, Inc.
31520 Beck Road
Bulverde, TX 78163-3911
USA

Phone: +1 (830) 438-4830

E-mail: info@rbc-us.com

Web: www.rbc-us.com

Twitter: @RBCS, @LaikaTestDog

Facebook: RBCS-Inc.

www.rbc-us.com