

IBM Software Group

P17 System Testing

Monday, September 24, 2007

Module 2: Introduction to Test Levels and Triggers

Rational software



Marty Swafford

IBM Rational Software

IBM Certified Solution Designer - Rational Manual Tester, Rational Performance Tester, Rational Functional Tester for Java

mswaffor@us.ibm.com

Go out looking for one thing, and that's all you'll ever find.”

Old prospector's saying



Module Objectives

After completing this module, you will be able to:

- Define the levels of testing
- Explain how the test planning for test levels relates to the overall development process
- Understand the concept of triggers

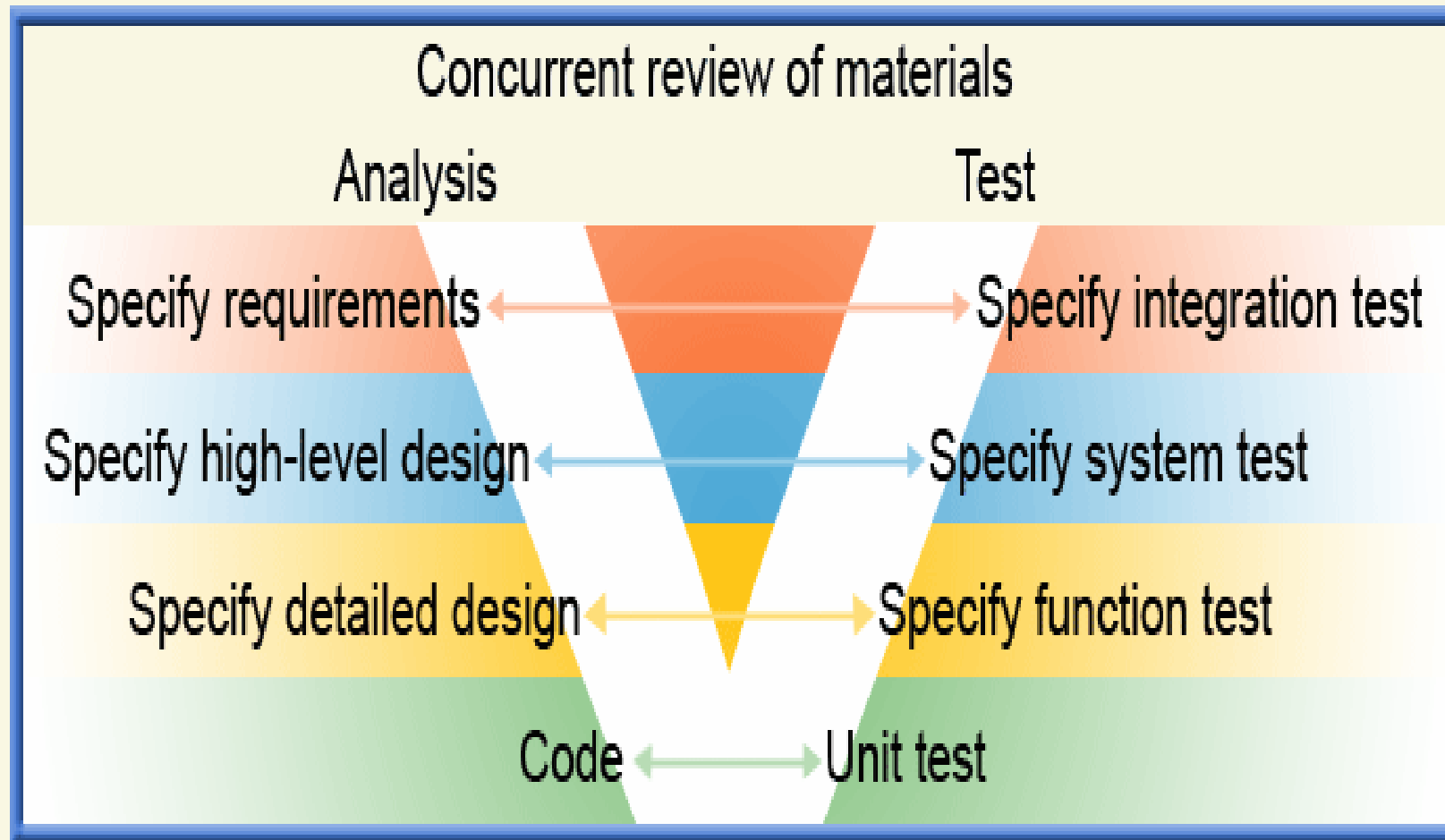


Test Levels

Test Level	Description
Unit Testing	Unit test verifies the functioning, in isolation, of software pieces that you can test separately.
Functional Testing	Functional testing verifies that each function of the software solution operates in accordance with the written requirements specifications.
System Testing	System testing is the verification of an entire solution, or set of solutions, in predefined system environments.
Integration Testing	Integration testing covering the broadest range of requirements and functionality for the whole system in the production environment. It also includes checking that the solution works with other products.
Acceptance Testing	Acceptance testing is to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system



Test Planning and the V-Model



Triggers

Triggers are the conditions or environment under which a defect surfaces.

Understanding the kinds of triggers relevant to each test level enables you to do the following:

- Use your knowledge to ensure that you test for each of the different types of triggers for the test level.
- Identify which classes of defects might escape to the next level of testing if adequate testing is not performed in the current level.
- Explain the repercussions of not covering all triggers that apply to a given level.



Orthogonal Defect Classification (ODC)

ODC is a scheme to capture the semantics of each software defect quickly. It is the definition and capture of defect attributes that make mathematical analysis and modeling possible.

Analysis of ODC data provides a valuable diagnostics method for evaluating the various phases of the software life cycle (design, development, test and service) and the maturity of the product. This is much like the diagnostics done in medicine using the blood sample from a patient to understand the existing health conditions and arrive at corrective actions.

ODC makes it possible to push the understanding and use of defects well beyond quality.



More ODC Info

- IBM Center for Software Engineering –
Orthogonal Defect Classification
<http://researchweb.watson.ibm.com/softeng/ODC/ODC.HTM>
- ODC Outside of IBM –
Books, Journals, White paper, URLs and Conference Proceedings
http://researchweb.watson.ibm.com/softeng/comm/odc_ext.htm



Module summary

- The levels of testing include unit testing, functional testing, system testing, integration testing and acceptance testing.
- Test and development planning *proceed simultaneously* and can be represented with the standard V-model.
- A trigger is the condition or environment that causes a defect to occur.

