



IBM Software Group

P17 System Testing

Monday, September 24, 2007

Module 6 : Testing Iteratively

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

If there is a 50-50 chance that something can go wrong, then 9 times out of 10 it will.”



Module Objectives

After completing this module, you will be able to:

- Define the Rational Unified Process (RUP)
- Define RUP for Testing Discipline
- Explain the RUP Test Discipline workflows



What Is the Rational Unified Process?



The Rational Unified Process (RUP)

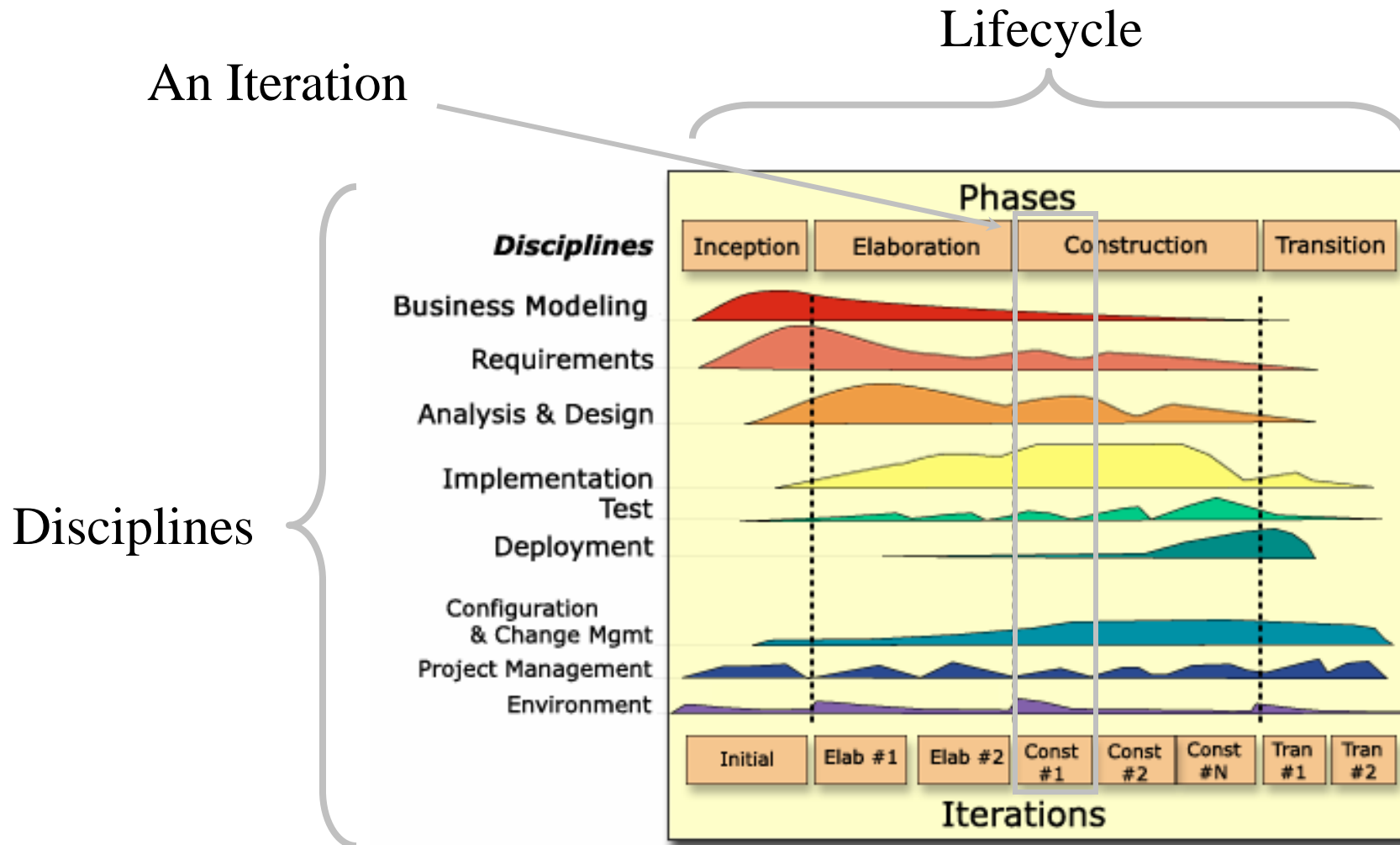
is an iterative software engineering process that provides a disciplined approach to assigning tasks and responsibilities within a development organization.

The iterative nature of **RUP** puts emphasis on the early initiation of testing, providing concrete feedback on progress, correctness and quality; allowing management teams to make timely decisions on corrective action.

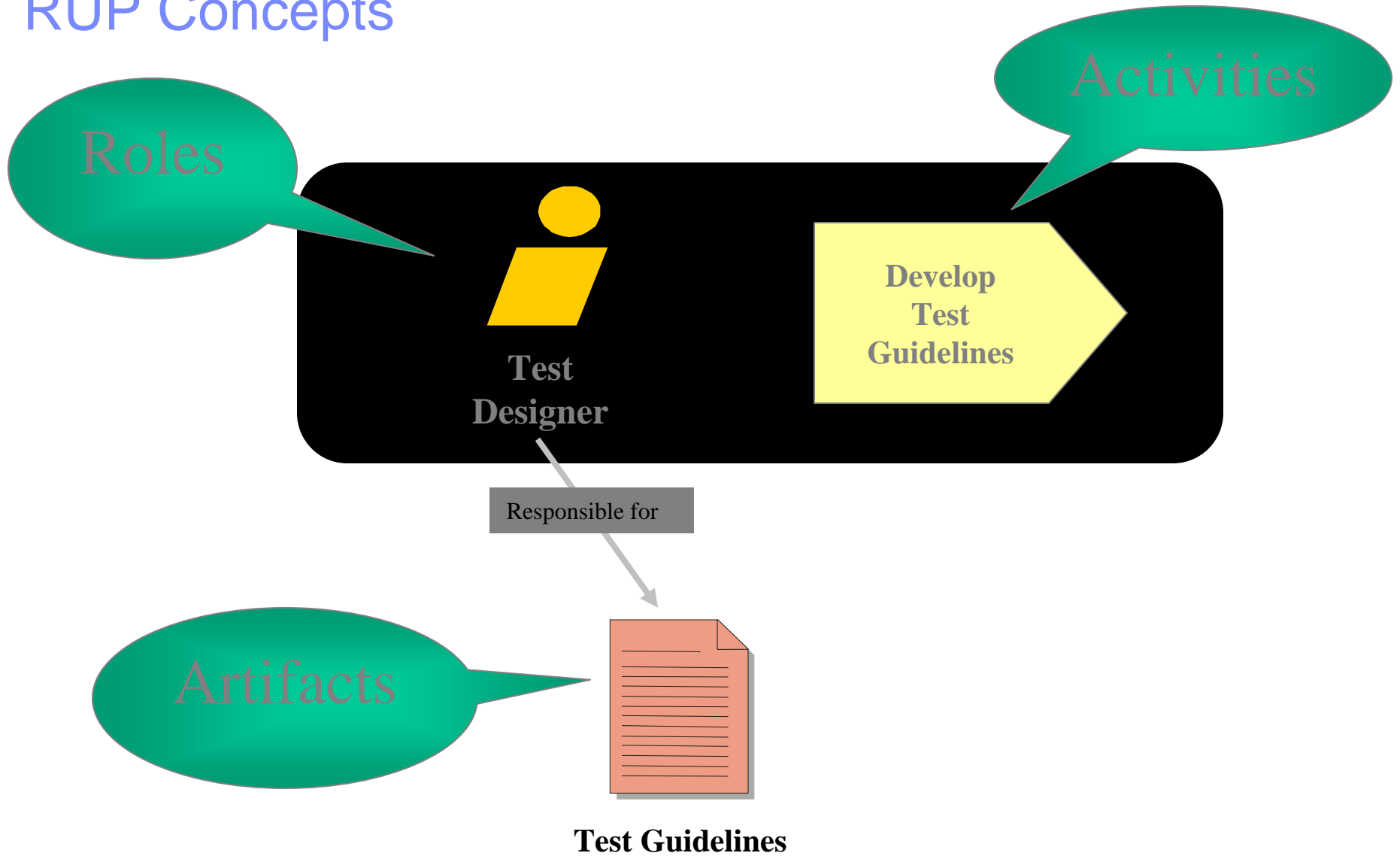
RUP's goal is to ensure the production of high-quality software that meets the needs of its users, within a predictable schedule and budget.



RUP Process Architecture

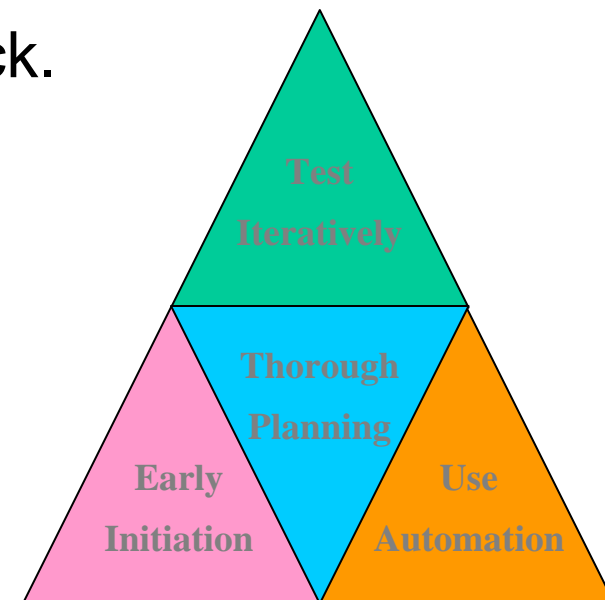


RUP Concepts

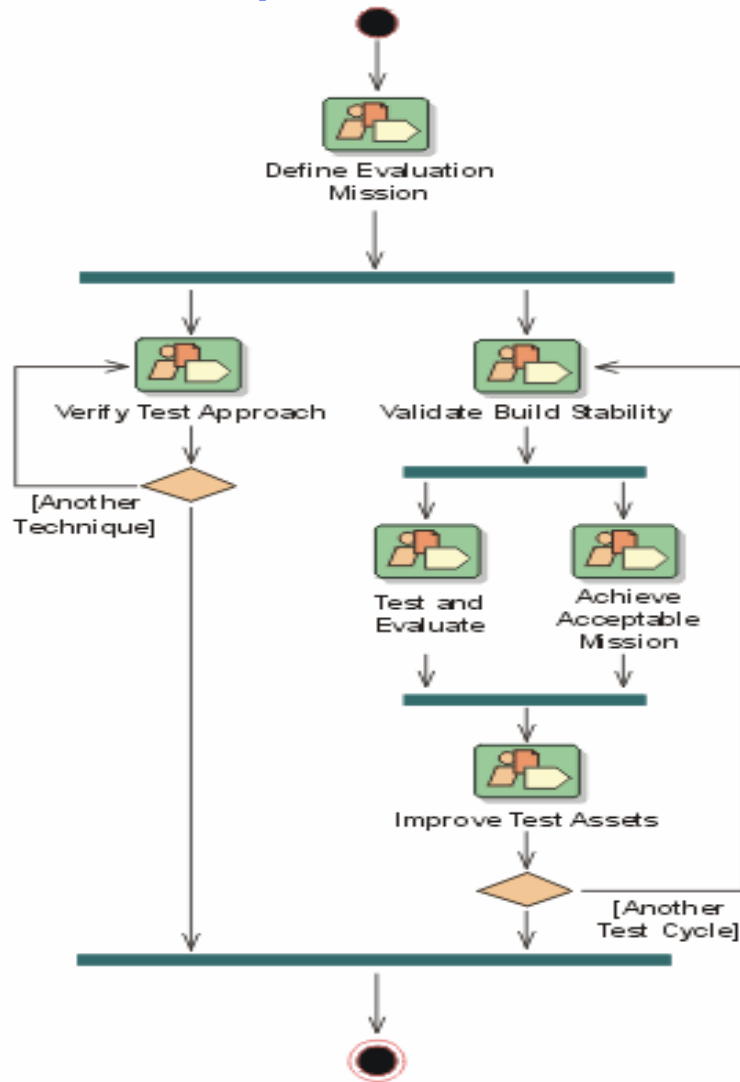


RUP Test Discipline

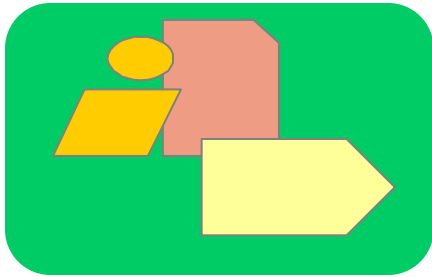
- Systematic approach to ensure testing effectiveness and efficiency.
 - ▶ Advocates iterative testing process.
 - ▶ Produces early concrete feedback.
 - ▶ Scalable and Customizable.
 - ▶ Designed for flexibility.
 - ▶ Focused on efficiency.



The RUP Test Discipline Workflow



Define Evaluation Mission

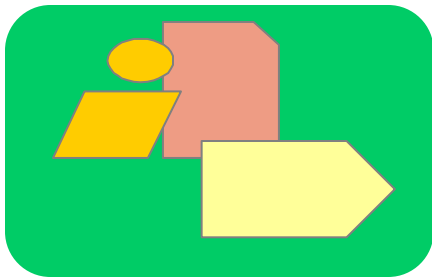


**Define Evaluation
Mission**

For each *iteration*:

- Identify the objectives for and deliverables of the testing effort.
- Identify a good utilization strategy for test resources.
- Define the scope and boundaries for the test effort.
- Outline the approach that will be used.
- Define how progress will be monitored and assessed.

Verify Test Approach



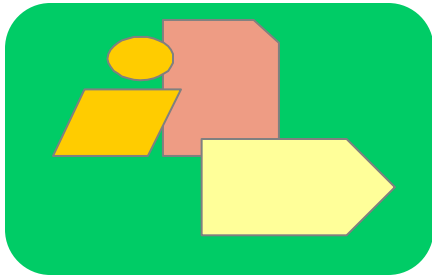
Verify Test Approach

For each *technique*:

- Verify early that the intended test approach will work and confirm its value.
- Establish the basic infrastructure to enable and support the approach.
- Obtain commitment from the development team to provide and support the required testability to support the approach.
- Identify the scope, boundaries, limitations and constraints of each technique.



Validate Build Stability

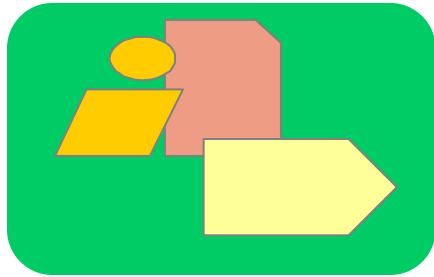


Validate Build Stability

For each *test cycle*:

- Assess the stability and testability of the Build.
- Gain an initial understanding, or confirm the expectation, of the development work delivered in the Build.
- Make a decision to accept the Build and to use in further testing, or to conduct testing against a previous Build.

Test and Evaluate



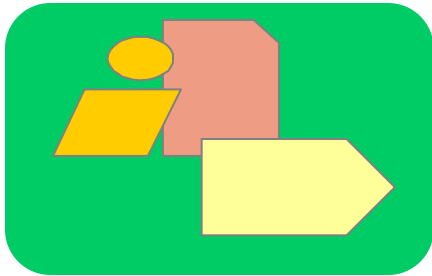
Test and Evaluate

For each *test cycle*:

- Provide ongoing evaluation and assessment of the Target Test Items.
- Record the appropriate information necessary to diagnose and resolve any issues identified.
- Achieve suitable breadth and depth in the test and evaluation work.
- Provide feedback on the most likely areas of potential quality risk.

Achieve Acceptable Mission

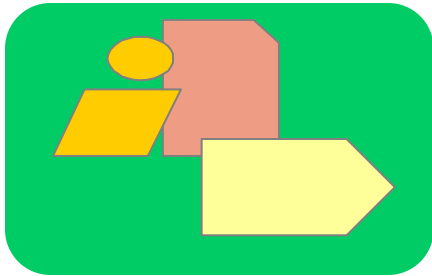
For each *test cycle*:



**Achieve Acceptable
Mission**

- Actively prioritize the minimal set of necessary tests that must be conducted to achieve the Evaluation Mission.
- Advocate resolution of important issues that have significant negative impacts on the Evaluation Mission.
- Advocate appropriate quality.
- Identify regressions in quality introduced between one test cycle and the next.
- Revise Evaluation Mission as necessary.

Improve Test Assets



Improve Test Assets

For each *test cycle*:

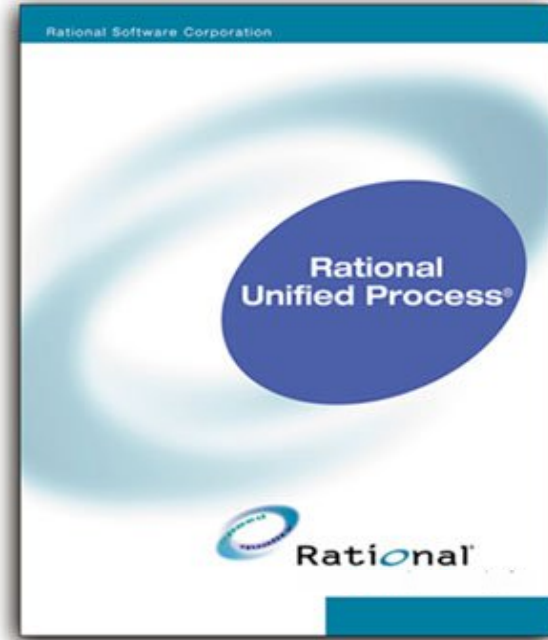
- Add the minimal set of appropriate additional tests to the smoke test.
- Assemble new test scripts into appropriate test suites.
- Remove unproductive or uneconomic test assets.
- Maintain test environment configurations and test data sets.
- Conduct general maintenance and maintainability improvements.
- Document lessons learned.

For Effective and Efficient Testing

Tools

+

Process



Module summary

- The Rational Unified Process is an iterative software engineering process that provides a disciplined approach to assigning tasks and responsibilities within a development organization.
- RUP simply consist of roles, activities and artifacts.
- RUP Test Discipline is a systematic approach to ensure testing effectiveness and efficiency.
- RUP Test Discipline Workflow consist of:
 - ▶ Define evaluation mission
 - ▶ Verify test approach
 - ▶ Validate build stability
 - ▶ Test and evaluate
 - ▶ Achieve acceptable mission
 - ▶ Improve test assets

