

# Mosaic’s Testing Process (MSTAR™) And the Rational Unified Process (RUP®)

## A Powerful Union for Quality Systems

Mosaic’s Structured Testing and Assessment Repository (MSTAR™) is a proven testing process providing state-of-the-art testing processes. The Rational Unified Process (RUP®) is a comprehensive software development process incorporating state-of-the-art development processes. Much interest has been expressed on how these two powerful methodologies fit together. This paper describes how MSTAR™ complements and extends the power of RUP®.

### Outline

Outline .....	1
Executive Summary .....	1
The Rational Unified Process (RUP®) .....	4
RUP® Testing and Risk Management .....	5
Mosaic’s Structured Testing and Assessment Repository (MSTAR™) .....	6
MSTAR™ and RUP® .....	6
MSTAR™ and RUP® Testing Stages .....	7
MSTAR™ and RUP® Plan Test Activities .....	7
MSTAR™ and RUP® Design Tests Activities .....	8
MSTAR™ and RUP® Implement Test Activities .....	9
MSTAR™ and RUP® Execution and Evaluation Activities.....	9
MSTAR™ and RUP® Test Management Activities.....	10
Conclusion .....	10

### Executive Summary

RUP® places a high value on testing early and continuously to measure and provide feedback on the developed product’s quality. RUP® includes a testing discipline that is responsible for iteratively performing the required testing. This discipline is supported by quality concepts and founded on industry-recognized testing activities. While the RUP® testing discipline is a good model for quality testing, it does not provide the depth of testing guidance needed to ensure that testing consistently meets its quality objectives across all projects.

MSTAR™ shares RUP®’s vision of early and continuous testing and is supported by similar quality concepts. MSTAR™ encompasses similar industry-recognized testing activities but since it is based on years of experience focused on developing testing expertise, it offers a depth of guidance, best practices and support that transforms

RUP<sup>®</sup>'s testing discipline into a comprehensive, consistent and highly effective risk management tool. The following table summarizes the testing support provided by RUP<sup>®</sup> and MSTAR<sup>™</sup> for the key testing activities.

Testing Activities	RUP <sup>®</sup>	MSTAR <sup>™</sup>
Testing Organization	<ul style="list-style-type: none"> <li>• Four Test Stages Identified               <ul style="list-style-type: none"> <li>○ Unit</li> <li>○ Integration</li> <li>○ System</li> <li>○ Acceptance</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Test Levels Defined Based on Responsibilities</li> <li>• Standard Test Levels Defined               <ul style="list-style-type: none"> <li>○ Unit</li> <li>○ System</li> <li>○ Acceptance</li> <li>○ Pilot</li> </ul> </li> <li>• Customized Test Levels Supported Based on Project Needs               <ul style="list-style-type: none"> <li>○ Integration Test Level is a Common Custom Test Level</li> </ul> </li> </ul>
Plan Test	<ul style="list-style-type: none"> <li>• One Test Plan Template/Sample for All Test Stages above Unit Test</li> <li>• Test Plan Includes High Level Requirements, Types of Testing and Testing Resources/Tasks</li> <li>• No Explicit Unit Test Planning Activity</li> </ul>	<ul style="list-style-type: none"> <li>• Multiple Test Plans               <ul style="list-style-type: none"> <li>○ Master Test Strategy – Overall Plan to Establish Foundation for All Test Levels</li> <li>○ Detailed Test Plan(s) – Test Plan for Each Test Level above Unit Test</li> </ul> </li> <li>• Comprehensive Test Planning Activities/Templates and Samples Provide a Solid Foundation for Testing Based on Risk</li> <li>• Streamlined Unit Test Plan</li> </ul>
Design Tests	<ul style="list-style-type: none"> <li>• Test Cases/Test Procedures Developed Based on Types of Testing Required and High Level Requirements</li> </ul>	<ul style="list-style-type: none"> <li>• Super Scenarios, Scenarios and Test Cases Developed Based on Testable Requirements and Proven Testing Techniques</li> <li>• Test Coverage Managed through Traceability of Scenarios to Requirements</li> <li>• Different Types of Testing Included Based on Need to Cover Different Types of Requirements</li> <li>• Data Managed Outside Scenarios in Data Profiles to Promote Reusability</li> </ul>

Testing Activities	RUP <sup>®</sup>	MSTAR <sup>™</sup>
Implement Test	<ul style="list-style-type: none"> <li>• Emphasis on Automation through Development of Test Drivers and Test Scripts</li> </ul>	<ul style="list-style-type: none"> <li>• Emphasis on Test Automation Where Cost Effective</li> <li>• Automation Strategy Developed to Ensure Automation Accomplishes Testing Objectives</li> <li>• Test Driver and Test Scripts Developed Based on Automation Strategy</li> <li>• Guidelines for Manual Tests that Support and Supplement Automation</li> </ul>
Execute Tests	<ul style="list-style-type: none"> <li>• Defined Tests Executed</li> </ul>	<ul style="list-style-type: none"> <li>• Defined Tests Executed</li> <li>• Guidelines/Templates/Samples Included for: <ul style="list-style-type: none"> <li>○ Test Schedules</li> <li>○ Execution Checklists</li> <li>○ Test Execution Logs</li> </ul> </li> </ul>
Evaluate Tests	<ul style="list-style-type: none"> <li>• Problems Entered as Change Requests</li> <li>• Concepts Outline the Need for Coverage, Defect Management and Test Execution Statistics</li> </ul>	<ul style="list-style-type: none"> <li>• Problems Entered as Defects</li> <li>• Guidelines/Templates/Samples Included for: <ul style="list-style-type: none"> <li>○ Defect Reports</li> <li>○ Test Coverage Reports</li> <li>○ Test Status Reports</li> </ul> </li> <li>• Objective Coverage Measures Supported through Process for Sizing Software Based on Testable Requirements</li> </ul>
Test Management	<ul style="list-style-type: none"> <li>• Work Plan Prepared as Part of Project Management Discipline</li> <li>• Various Size Measurements and Testing Metrics Conceptually Suggested</li> </ul>	<ul style="list-style-type: none"> <li>• Guidelines/Template for Testing Work Plan Provided</li> <li>• Testing Estimating Guidelines Provided</li> <li>• Process for Estimating System Size Based on Testable Requirements Provides Consistent Size Measurement Throughout the Development Life Cycle</li> <li>• Sample Reports Provide Testing Metrics</li> </ul>

As illustrated in the previous table, MSTAR™ and RUP® support the same quality testing workflow, but MSTAR™ completes and extends the RUP® process with comprehensive guidelines, templates and samples based on industry testing experience.

## ***The Rational Unified Process (RUP<sup>â</sup>)***

The Rational Unified Process (RUP®) is a software development process from Rational Software that embodies state-of-the-art approaches for developing systems including:

- Object technology and component-based development,
- Modeling using the Unified Modeling Language (UML),
- Architecture,
- Iterative development,
- Requirements management,
- Change management and
- Verification of software quality.

RUP® is documented in a browser-based repository available on the project team's workstations. Rational Software is continuing to evolve the process providing a living development life cycle. The development process is supported by an array of tools provided by Rational Software.

In contrast to traditional waterfall approaches, RUP® breaks the development cycle into a succession of iterations. Each iteration involves a mini waterfall process including requirements, design, implementation and testing. To manage the project, the development cycle is divided into four phases that group the sequence of iterations. These phases are:

- **Inception:**  
Defines the project culminating with an end product vision that represents concurrence among all stakeholders on the objectives for the project.
- **Elaboration:**  
Analyzes the problem domain to establish a sound architectural foundation and reliable development plan for the project. Elaboration must stabilize the requirements and architecture to a point that mitigates the highest-risk elements of the project.
- **Construction:**  
Builds the product and evolves the vision to produce an initial operational release. All remaining components and system features are incrementally developed, integrated and verified. Management focuses on optimizing costs, schedules and quality.
- **Transition**  
Deploys the system into the user environment completing final verification and changes to result in the product release.

## RUP® Testing and Risk Management

Successful implementation of RUP® requires strong risk management. The high-risk areas must be identified early and the Elaboration phase must stabilize the baseline architecture and requirements to eliminate identified high risks. Remaining risks must then be managed to allow the iterative development to progress in a controlled manner to meet its cost, schedule and quality objectives.

A key risk management technique embraced by the Rational process is the testing discipline. Within RUP®, the testing discipline is an iterative discipline occurring in all phases to provide early and continuous feedback on product quality. It is a vital part of the project's ongoing feedback mechanisms, enabling quality to be measured and change requests to be identified.

The RUP® testing discipline is composed of the following activities:

- **Plan Test:**  
Creates a test plan identifying the requirements that will be tested, the types of testing that will be performed and the associated testing resources/tasks.
- **Design Test:**  
Defines the actual test cases and test procedures that will be executed according to the test plan.
- **Implement Test:**  
Develops the test scripts to accomplish the planned testing.
- **Execute Tests in Integration Test Stage:**  
Executes the defined test scripts and manual testing for Integration Testing.
- **Execute Tests in System Test Stage:**  
Executes the defined test scripts and manual testing for System Testing.
- **Evaluate Test:**  
Evaluates the test results, prepares change requests and generates measures to assess the product quality. Measurement includes code and requirement coverage as well as defect management and test execution statistics.

These activities are repeated in an iterative fashion, as appropriate, through all phases of the process.

While the defined RUP® testing discipline, performed in an iterative manner throughout the life cycle, is a valid testing model, it does not provide the comprehensive guidelines, techniques and tools available with current state-of-the-art testing best practices. Mosaic's Structured Testing and Assessment Repository (MSTAR™) can complete and extend RUP®'s testing model with proven testing practices to meet the risk management and quality objectives of the Rational process.

## ***Mosaic's Structured Testing and Assessment Repository (MSTARÔ)***

MSTAR™ is a browser-based repository of testing expertise. It provides an online reference to a proven testing process and includes supporting samples, templates and forms. Mosaic's testing methodology is the most advanced in the industry. It has been field-proven on mainframe, client/server, PC and Web systems, and is applicable to custom-developed systems, package implementations, and out-sourced development.

MSTAR™ embraces and further develops the concepts embodied in RUP® including:

- **Risk Management:**  
Testing resources are focused where they will do the most good.
- **Early Test Planning:**  
Test planning can begin as soon as the project is initiated.
- **Incremental Testing:**  
Components are delivered and tested incrementally according to a build strategy.
- **User Involvement:**  
Users are integrated into the testing process.
- **Test Automation:**  
Test Automation is planned and applied, where appropriate, to improve testing efficiency.
- **Measurement:**  
Key measures including system size, planned and completed coverage, and defects are used to quantify the progress and identify risk.
- **Management Reporting:**  
Management is kept informed of progress, issues, and cost/coverage tradeoffs.
- **Defect Analysis:**  
Defect information is used to align coverage strategies with risk, predict schedules and gain insight into development issues and problems.

### ***MSTARÔ and RUPâ***

MSTAR™ and the Rational process both embrace the concept of early and continuous testing to measure and monitor system quality and manage risk. MSTAR™ completely supports the documented RUP® testing discipline with a similar workflow and consistent quality concepts. MSTAR™, however, further extends the RUP® testing discipline with testing best practices that transform the RUP® testing discipline into a comprehensive and highly effective testing process.

MSTAR™ also is highly compatible with RUP® in its structure and design. Both products are browser-based and can thus be easily linked together using the power of Web-based applications. Similar to Rational's commitment to RUP®, Mosaic is continuing to develop MSTAR™ making it a living testing process. MSTAR™ is compatible with all standard development and testing tools.

## **MSTAR™ and RUP® Testing Stages**

RUP® refers to four stages of testing:

- Unit
- Integration
- System
- Acceptance

All four stages follow the same iterative workflow.

MSTAR™ supports and extends this concept. MSTAR™ defines test levels (analogous to test stages) as progressively higher stages of testing that are differentiated by responsibility. Whenever the responsibility or testing role assignment changes, a new test level is defined so the movement between the test levels can be controlled through entry/exit criteria and through clear differentiation of responsibilities. MSTAR™ contains four standard test levels:

- Unit
- System
- Acceptance
- Pilot/Parallel

MSTAR™ also supports customization of test levels based on project risk. An Integration test level is an example of a custom test level. Thus MSTAR™ supports the test stages within RUP® but extends the concept through the flexibility of customized test stages based on the project and through more extensive test planning that defines specifically how testing will be spread across the test levels based on the needs of the specific project.

## **MSTAR™ and RUP® Plan Test Activities**

RUP® defines a Plan Test activity and a Test Plan template that are used for all test stages. The Test Plan covers the high level requirements that must be tested, the types of testing required, the testing resources and roles needed, and the testing milestones.

MSTAR™ supports and extends this concept by providing guidelines and templates/samples for different types of test planning. While each of the test stages or levels must plan their own testing, there are many benefits to performing central planning to address issues that affect all test stages/levels. There are many testing foundational issues that can most efficiently be addressed centrally. Also, the different stages/levels can more clearly fulfill their responsibilities if the testing responsibilities are clearly communicated to all involved individuals. MSTAR™'s central test plan is called a Master Test Strategy.

The Master Test Strategy addresses the overall plan for testing on the project. It is a key test plan that is developed early to assess the project risk and plan the testing that will be required across the entire project. The Master Test Strategy includes a definition of the test levels and planning for the many testing concerns that cross the test levels including test environment, test data, test control procedures, test tools, regression test strategy, etc. While some of these concerns are covered in RUP®'s Environment, and Configuration and Change Management disciplines, this artifact focuses the test planning on concerns key to its success. It provides the input to these other RUP® disciplines, as well as extending the planning within each discipline to ensure a solid foundation for testing.

Beyond the Master Test Strategy, MSTAR™ provides guidelines and tools to define detailed test plans for system test and other test levels. These test plans provide the detailed strategies to guide the testing at those test levels. Finally, unit test plans are supported separately since this test level, while similar to the other test levels, often requires a streamlined format. Unit testing is required by the Rational Process but unit test planning is not detailed as a specific activity. MSTAR™ provides specific guidelines for this test level recognizing its importance as an early and cost-effective test level.

In addition to providing multiple types of test plans, MSTAR™'s test plans also provide more comprehensive test planning covering all aspects of the required testing. All test plans detail the testing required based on project risk. Sample areas covered in addition to those included in RUP® are test data strategies, test automation plan, regression test strategies and coverage strategies. MSTAR™'s approach to test planning provides a solid foundation for quality testing based on project risk.

## **MSTAR™ and RUP® Design Tests Activities**

The RUP® Design Tests activity identifies tests based on the types of testing required and the high level requirements to be tested. This approach will generate a valuable set of test cases; but to ensure adequate coverage and risk management, MSTAR™ plans testing based on high level requirements that have been decomposed to the detail or testable requirements level, as communicated in the use cases, other requirements and design models. Decomposing requirements to the testable requirements level ensures that each requirement is clearly understood and that the requirement coverage by the planned test cases is clear and measurable. All types of testing are covered by this test planning approach since the different types of testing listed in RUP® represent the different types of requirements that the test planning must ensure receive adequate coverage.

MSTAR™ also enhances the RUP® Design Test activity by providing additional guidelines on planning the specific tests required and on measuring coverage based on testable requirements. RUP® supports this approach in its measurement and best practices concepts but not in its activities and deliverable standards.

Finally, MSTAR™ manages test data separately from the test scenarios through the use of data profiles. Defining the test data outside the test scenarios in data profiles enables the creation of highly reusable test scenarios. This approach also facilitates test automation, which is most successful when the defined scripts are driven by outside data.

## **MSTAR™ and RUP® Implement Test Activities**

In RUP®, the Implement Test activity covers the development of the test scripts and test drivers required for automated testing. MSTAR™ also places emphasis on automated testing, but it focuses on automation that is cost-effective for the current project. MSTAR™ creates an automation strategy to ensure that the automation efforts are focused and structured to meet the long-term requirements for testing the system, and to ensure that resources and test tools are consistent with the approach. Test scripts and test drivers are then defined based on the automation strategy. In addition to the test automation, MSTAR™ also provides support for the development of the manual test cases that support and augment the test automation.

## **MSTAR™ and RUP® Execution and Evaluation Activities**

The Execution activities are the same in RUP® and MSTAR™ except MSTAR™ is generalized to cover execution at all test levels. MSTAR™ and RUP® provide similar guidelines for evaluating and measuring the test results. Although MSTAR™ again extends the process by providing samples and templates for:

- Test Execution Schedules
- Execution Checklists
- Test Execution Logs
- Defect Reports
- Test Coverage Reports
- Test Status Reports

MSTAR™ also provides a process for measuring system size in terms of testable requirements. This provides a powerful tool for facilitating the size measurement referred to in RUP®. A testable requirements size measure also greatly increases the objectivity of the coverage measures.

## **MSTAR™ and RUP® Test Management Activities**

RUP® includes test management as part of its Project Management Discipline. Testing is included in and managed through the project management artifacts. Conceptually, RUP® recommends various metrics to measure the system size, and monitor quality and risk.

MSTAR™ supports RUP®'s goal to manage the risk and optimize the productivity of a project. It extends RUP®'s process by providing guidelines, samples and templates focused on managing the testing effort. A testing work plan template and estimating guidelines provide detailed support for the generation of the testing project plan. MSTAR™ also enhances RUP®'s risk management capabilities by supporting the management of the testing effort to maximize the reduction of risk on the project.

MSTAR™ also supports and enhances RUP®'s process to provide meaningful metrics. MSTAR™'s process for sizing the system based on testable requirements provides a powerful and consistent method for measuring the size of the system throughout the development life cycle. The size measurement can assist with estimating work efforts, risk assessment and measurement of system change, productivity and coverage. Report templates and samples within MSTAR™ also provide tools to create testing metrics to monitor quality and risk.

### ***Conclusion***

RUP® provides state-of-the-art guidelines for developing today's mission critical systems. It places high value on testing early and continuously to measure and provide feedback on the developed product's quality. MSTAR™ is a proven testing process providing state-of-the-art testing processes. MSTAR™ supports the same quality testing process included in RUP® but it extends that testing process with proven testing techniques and best practices. Together MSTAR™ and RUP® provide a powerful process for developing quality systems in today's fast paced and high-risk environments.