

Perform integration testing earlier

*Virtualize WebSphere MQ endpoints for higher quality
and lower project risk*



The challenges of integration testing today

IBM® Rational® test automation solutions are designed to help development teams virtualize WebSphere® MQ endpoint dependencies for earlier testing in the software development cycle. In turn, this early virtualization enables continuous integration and testing to identify and resolve defects as early as possible.

In a world of fast-paced development and increasing reliance on agile development methods, development teams must meet the needs of modern businesses by delivering innovative, high-quality software at an accelerated rate. To achieve high-quality results, it is necessary to test all layers of the application much earlier and throughout the development cycle, and this can be very difficult in the complex, heterogeneous environments that host composite applications. In fact, **testing has become a major bottleneck in application development.**

In an effort to address this development bottleneck and improve the overall quality of their software, development teams often strive to include continuous integration testing as part of their automated process. However, they frequently fall short of their desired testing objectives because it is difficult, expensive and time-consuming to set up an integrated testing environment. If continuous integration testing is impractical, teams often rely on less comprehensive testing methods such as “continuous compilation” or “continuous unit testing.”

Today’s development projects typically involve enterprise application integration (EAI), service-oriented architecture (SOA), event-driven architectures (EDA) and distributed systems interconnected by an enterprise service bus that is implemented using an IBM WebSphere MQ solution. There are usually no human interfaces for these interconnected applications or for the

protocols (for example, SOAP or XML) used to distribute information between machines. **Traditional approaches to test automation that focus on recording and playback of the user interface are impractical.**

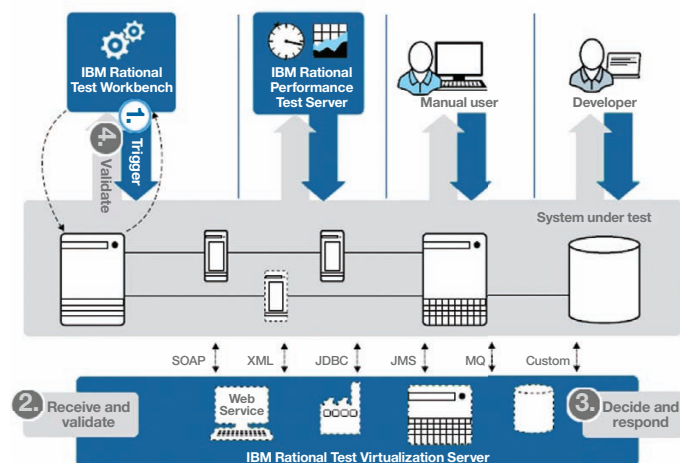
Many of today’s test automation solutions require the test professional to sit in front of a computer and use the mouse and keyboard to manually interact with the application and generate a test script for automated regression testing. These automated scripts then require ongoing maintenance as the application’s user interface changes. As a result, test automation is sometimes deferred until the user interface has stabilized. **Delayed test automation defers risk to late stages of a project.** In addition, graphical user interface (GUI) testing does little to assist with isolating defects or accelerating defect resolution in composite application deployments.

Addressing challenges with Rational test automation solutions

A user-driven approach to testing is no longer sufficient. To decrease the cost of high-quality results, today’s test teams must ensure that the development process includes earlier automated testing of the integration between all dependent software and services. This is where IBM can help. The capabilities provided by IBM Rational test automation solutions can help development and test teams virtualize WebSphere MQ endpoint dependencies earlier in the development cycle, thereby facilitating continuous integration and testing. Some of these dependencies will be owned or delivered by teams with their own projects and pressures. Even if a component is being developed as part of this project, there might well be a need to test components that depend on it in isolation or even before the new component has been completed.

Automated test functionality with IBM Rational test automation solutions

- 1) The tester sends a message “trigger” to a service in the system under test, which in turn needs to exchange information with another service.
- 2) Test virtualization capabilities provide a virtual component to satisfy the service dependency; validating requests are received.
- 3) The virtual component provides a response to the message, such as error, look up request or timeout.
- 4) The system responds to the original request and validates whether the response is correct.



Emulating missing dependencies for the purpose of testing has been called service virtualization by some industry analysts. However, such emulation can use non-service based approaches, and IBM believes the term test virtualization is more appropriate. In test virtualization, a real component is replaced by a virtual component, sometimes called a stub. Virtual components should be made available for key services to enable various

situations to be simulated and tested more easily. If key services are delivered late, the availability of virtual components ensures that problems in dependent components can still be detected, and the project can “flow around” the blockage.

Rational test automation solutions include two key capabilities that are designed to help developers more effectively test applications that rely on WebSphere MQ:

- Virtualize or simulate dependent software and services.
- Record or author test cases of these virtualized or readily available dependent services.

After a virtualized component is deployed, it monitors the appropriate WebSphere MQ message queue for incoming requests. If it receives a request, the virtual component can post appropriate responses based on the message format and content of the request. There are two benefits to recording or authoring tests at the service interface layer:

- This approach to automated testing requires less test script maintenance because the service interface layer should be well documented and is usually more stable.
- Automating integration at the service layer enables your development and test teams to validate layers of the application and to automate testing of core functionality before the GUI is available.

One common challenge in testing composite applications is being able to simulate situations that only reveal themselves as problematic after all of the “piece parts” or project components are integrated and functioning as a cohesive application. For example, have you ever tried to measure how many transactions you can drive through a specific component or to understand how an application will react to a transaction that runs for long periods? Without early proactive testing to emulate the condition of a long-running response, you risk finding out about problems only after the composite software is deployed to a production environment.

Rational test automation solutions can help your development and test teams address performance concerns in two ways:

- 1) Virtual components can be configured to limit throughput, which allows you to measure the impact long-running transactions on other components.
- 2) Performance constraints can be lifted from virtual components to measure the maximum throughput you can achieve through your components.

IBM WebSphere MQ is a valuable middleware layer that coordinates functionality between complex systems and brokers communication between the integrated applications. If you are able to validate such integration earlier in the development lifecycle, you might be able to decrease the cost of resolving expensive defects, avoid late stage integration issues and reduce the overall cost of quality.

Conclusion

Rational test automation capabilities can be part of a capable and cost-effective testing approach. Rational test automation solutions currently support WebSphere MQ and more than 70 additional protocols and message formats, meaning that you can virtualize dependent software and services and validate applications where it matters—at the integration layer. This capability helps give you the option to implement earlier automated testing, which in turn can help reduce cycle times, business risk and the cost of software quality.

For more information

To learn more about test automation solutions from IBM, contact your IBM marketing representative or IBM Business Partner, or visit the following website:

ibm.com/software/rational/offerings/quality

See also:

- IBM Rational Test Workbench
ibm.com/software/rational/products/rtw
- IBM Rational Performance Test Server
ibm.com/software/rational/products/rpts
- IBM Rational Test Virtualization Server
• ibm.com/software/rational/products/rtvs



© Copyright IBM Corporation 2012

IBM Corporation
Software Group
Route 100
Somers, NY 10589

Produced in the United States of America
July 2012

IBM, the IBM logo, ibm.com, Rational, and WebSphere are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at ibm.com/legal/copytrade.shtml

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

It is the user's responsibility to evaluate and verify the operation of any other products or programs with IBM products and programs. THE INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.



Please Recycle
