

# Application Management for Changing Times: Micro Focus Service Test Management

The Micro Focus® Service Test Management module enables QA teams to manage components of an application under test while keeping in step with development.

## Modern QA for Modern Applications

Organizations continue to improve their responsiveness to market demands by investing in IT costs to modernize their applications. In a recent study of more than 200 companies worldwide, Forrester Consulting found that “69% of respondents have earmarked more than 25% of their IT budgets for modernization, while one-third of survey respondents will dedicate 51% of their IT budgets for modernization.”<sup>1</sup> This is a massive investment in the overhaul of inflexible legacy applications, and it will advance the adoption of Web 2.0 technologies, service-oriented architectures (SOA), agile and iterative development methodologies, and other strategies for increasing agility. It is imperative that QA teams update their practices to take on the challenges of testing these modern applications. Just as the development team is expected to leverage new technologies, shorten development cycles, and respond more quickly to business demands, QA must also become agile and deliver valuable services in this new era of application development.

Aging monolithic applications are being hidden behind a modern abstraction layer or

rewritten altogether with a component- or service-based architecture. A single modern application can have multiple components and layers, each with unique requirements, availability, and ownership. And with today’s tools, you no longer have to wait for a stable graphical user interface (GUI) to begin testing an application. Leveraging the power of Micro Focus Quality Center, the Service Test Management module can help you manage the quality of

these application components and services, and provide a stable foundation for your organization’s modern applications.

## Test Apps, Not Your Patience

Service Test Management extends Quality Center by providing management and information for composite applications. Access to this information allows QA to test earlier in the development lifecycle and execute only the

1 “Clearing Your Path to Modern Applications and Business Agility,” an April 2010 commissioned study conducted by Forrester Consulting on behalf of Micro Focus.

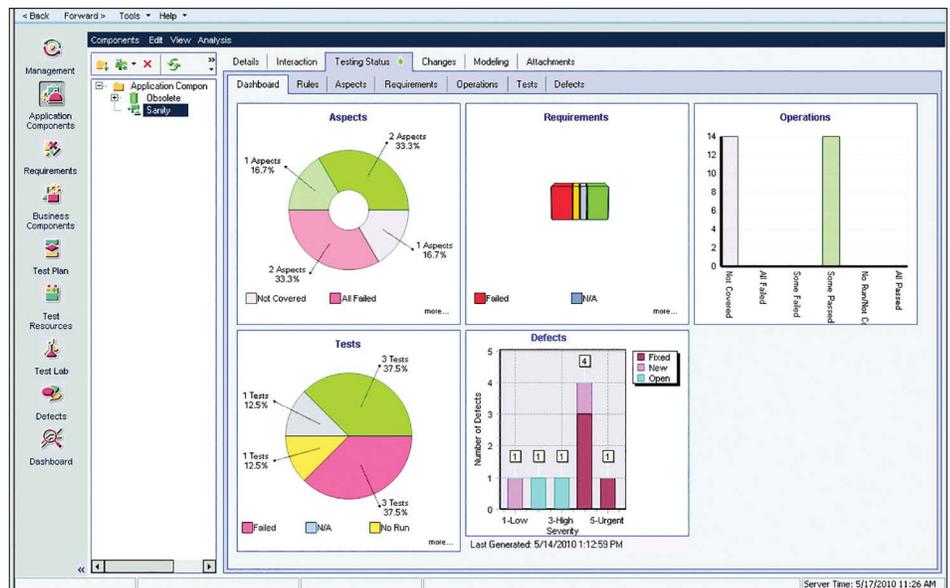


Figure 1. The Service Test Management dashboard provides quick, at-a-glance test coverage and testing progress for your application components.

tests needed, shortening test cycles and responding more quickly to change. With Service Test Management, your QA teams can:

- Participate effectively and add unique value to each iteration within an agile lifecycle
- Test earlier with a clear understanding of what components can and should be tested at different stages of the development lifecycle
- Manage and execute non-GUI testing (also known as “headless” testing), which is often the only option early in development before the GUI is stable
- Assess the impact of a change to an application component quickly, and test only what is affected, avoiding costly over-testing
- Determine test coverage and test status quickly at component and application levels

**Change Impact Analysis**

Service Test Management allows your QA team to represent dependencies between components within a composite application and in architectures where components are shared by multiple applications. Armed with this information, QA can quickly assess the impact on other components

and on consuming applications when a change occurs elsewhere in the system. When QA teams know exactly which components are impacted by a change, they can immediately begin testing affected areas, and can confidently avoid over-testing by excluding unaffected areas of the system—saving time and resources.

**Stay in Sync with Developers**

In the fast pace that is characteristic of agile development, QA teams are easily excluded. Short of embedding a test engineer in the cubicle of every developer, how can QA stay on top of the constant change? Service Test Management creates a platform for tracking modifications made to components by developers. This allows QA professionals to assess the impact to the overall system and quickly address affected areas within a sprint or iteration window.

**Best-in-Class Services Management**

Service Test Management plugs into Quality Center for service-oriented quality management. Services can be automatically discovered and downloaded from any Universal Description, Discovery, and Integration (UDDI)-compliant SOA system of record, such as SOA Systinet, through the UDDI interface or other mechanisms used to share services, such as URLs or files. Once the service is defined in

Quality Center, Service Test Management enables you to manage it alongside other application components. Quality Center can then report testing results and defects back into the application lifecycle through integration with SOA Systinet and drive critical lifecycle decisions—such as the decision to deploy a service into production—based on testing results.

**The Method for Test Management**

**Define a Component**

An application component is defined through a simple user interface (UI). If the application component type is not listed, Service Test Management is extensible and allows you to represent any new technology. In the case of services, you can import service information via Web Services Description Language (WSDL) through any UDDI repository, such as SOA Systinet.

**Create and Associate Quality Center Assets**

Once a component is defined, you can associate the component with the relevant tests and requirements. Service Test Management facilitates the generation of requirements and tests through a wizard.

**Represent Dependencies**

Service Test Management allows you to manage the dependencies between any application components. This information can be used to determine what testing needs to happen in the event of a change.

**Execute Tests**

Once the previous steps are completed, you have a library of test sets connected to your application components, and those tests can be executed as needed. Any defects discovered by the tests can then be linked to application components.

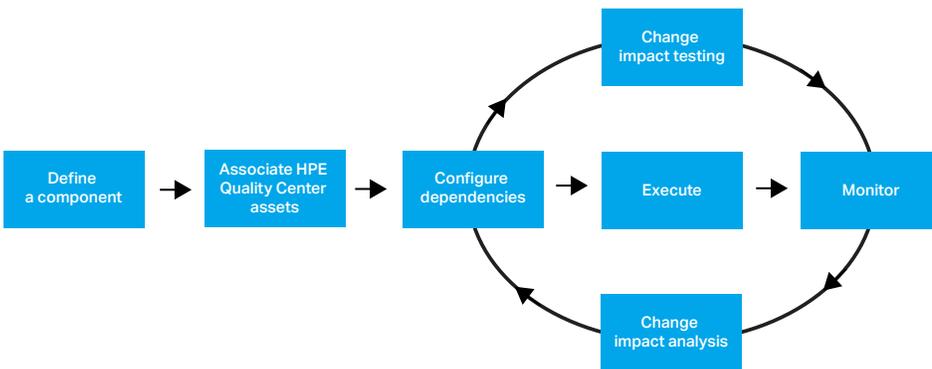


Figure 2. How Service Test Management works

## Monitor Application Quality

QA and development managers can review a summary of changes made to the application under test and of how application quality was affected. If a problem is detected, the development team can be notified and can respond.

## Change Impact Testing and Analysis

The QA team can review the application dependencies and determine which tests need to be run when a change occurs in development. Testers can then validate changes within a short test cycle by running only the tests that apply and excluding the tests that don't. After the cycle is concluded, the QA team can continue its ongoing monitoring of the system under test.

## Key Features and Benefits

- Automated detection of changes to application components
- Change-impact testing and the automatic creation of tests to validate change
- Easy test and requirements generation through a wizard
- Visual representations of the components of a composite application, and of the dependencies between those components
- Ability to apply quality rules and policies and alert the user if they are violated

## Quality Center

Service Test Management is part of Quality Center, an integrated set of capabilities and best practices for automating key quality activities, including requirements management, test management, defect management, functional testing, and business process testing. Using Service Test Management, you can associate all of these Quality Center assets with application components.

## Micro Focus Unified Functional Testing

Unified Functional Testing brings Micro Focus QuickTest Professional, QuickTest Professional add-ins, and Service Test together in a single package. This powerful combination of

functional testing tools allows you to conduct automated testing for both GUI-based applications and non-GUI-based services. Service Test is designed to work with Unified Functional Testing, and it enables you to design and execute tests for components managed by Service Test Management.

## A Complete Solution

### Comprehensive Training

Micro Focus provides a comprehensive curriculum of Micro Focus software and IT service management courses. These offerings provide the training you need to realize the full potential of your Micro Focus solutions, increase your network optimization and responsiveness and achieve a better return on your IT investments.

With more than 30 years of experience in meeting complex education challenges worldwide, Micro Focus knows training. This experience, coupled with unique insights into Micro Focus Software and Solutions products, positions Micro Focus to deliver an outstanding training experience.

### Micro Focus Professional Services

Get the most from your software investment.

Micro Focus provides high-quality software services that address all aspects of your software application lifecycle needs. With Micro Focus you have access to standards-based, modular, multi-platform software coupled with global services and support. The wide range of Micro Focus service offerings—from online self-solve support to proactive mission-critical services—enables you to choose the services that best match your business needs.

For an overview, visit: [www.microfocus.com/software/services](http://www.microfocus.com/software/services)

Learn more at  
[www.microfocus.com/functionaltesting](http://www.microfocus.com/functionaltesting)  
[www.microfocus.com/qualitycenter](http://www.microfocus.com/qualitycenter)

Contact us at:  
[www.microfocus.com](http://www.microfocus.com)

Like what you read? Share it.

