



WHAT YOU SHOULD EXPECT FROM LOAD TESTING YOUR WEB APPLICATIONS

How to Turn Your Test Results into Business Results

Quotium

EFFECTIVE PERFORMANCE MANAGEMENT

CONTENTS

TESTING COMPLEX WEB APPLICATIONS – A CRITICAL BUSINESS REQUIREMENT	2
RESULTS AND METRICS – WHERE IS THE VALUE?	2
WEB APPLICATION LOAD TESTING	3
THE RIGHT FEATURE MIX.....	5
LEVERAGING TEST DATA.....	7
LOWER COST OF OWNERSHIP – ONE TEST, VALUE FOR EVERYONE.....	7
GETTING THE RESULTS YOU NEED WITH QTEST	8
ANOMALY PROFILER	9
REPORT DESIGNER.....	10
CONCLUSION	11

TESTING COMPLEX WEB APPLICATIONS – A CRITICAL BUSINESS REQUIREMENT

According to a study by the National Institute of Standards and Technology (NIST),² annual costs to the country for inadequate application testing are as high as 59 billion dollars. As the adoption rate of mission critical Web-based applications continues to rise, poor testing represents a large portion of that cost. As a result, the ability to quickly and accurately test Web environments is not only a sound business decision, but it has become an absolute competitive necessity. Because Web applications are open to a large numbers of users, and typically allow direct access by prospects and customers, their functional efficiency and performance directly and immediately impact the bottom-line.

For example, a survey on E-commerce site performance, conducted by Jupiter Research for Akamai Technologies Inc. reveals that consumers will not wait more than 4 seconds for a Web page to download. One-third of shoppers who have a poor experience abandon the site entirely, while 75% are not likely to shop on that site again.

The decisions that IT leaders make in selecting a testing solution are, therefore, of high consequence. By nature, Web applications are complex and many involve proprietary software and third party applications interacting in complex multi-tiered architectures. For that reason, predicting, measuring and tuning the performance of Web applications pose unique testing challenges, and only a few enterprise-class Web testing solutions can address these challenges in their entirety.

Provided all enterprise-class solutions have comparable performance data gathering capabilities (there are some significant variations), much of the challenge resides in choosing the testing solution that will help make sense of the large amount of test-generated data. It should provide the analysis and reporting features that are most adapted to your needs in order to turn your test results into immediate business benefits. This paper identifies those few but essential analysis and reporting features that are critical to leverage test results and turn them into immediate business value for both the Executives and Engineers in your organization.

RESULTS AND METRICS – WHERE IS THE VALUE?

Test results hold a wealth of valuable and critical information for many members of your organization. For example, they might hold answers for an Executive as to whether an application will scale to support next year's projected growth. For an IT Manager, the same data set can indicate whether a new version of an application should be released, or where in the infrastructure upgrades will be necessary. For Developers, it can assist debugging the costly loss of performance in an application currently in production. The information is there, in a centralized repository, and beyond the high-level information on response time, it contains hundreds of counters that provide performance statistics about every Web and application server, operating system, network, database table, query, etc. However, what creates the value is the process of quickly extracting actionable information from this massive amount of data for the benefit of people everywhere in your organization.

So, what are the features that will enable you to deliver the right information in the right format to the right person? Lets look at some of the output and analysis features that will enable you to accurately pinpoint and quickly extract the specific data that each member of your organization needs.

² NIST, <http://www.itl.nist.gov/div897/docs/ConformanceAdvisory.html>

WEB APPLICATION LOAD TESTING

Load testing, in general, refers to the practice of modeling the expected usage of a Web application by simulating multiple users accessing an application's services and components concurrently. Across the professional software testing community, load testing is often used as a blanket term that is used in many different ways. Automated load testing is most relevant for multi-user systems and often, complex Web and multi-tiered architectures.

There are many approaches to automated load testing, but effective testing of complex Web-applications is essentially a process of testing an application's response time and limits while dynamically analyzing it and inspecting the results for issues. The specific requirements of load testing can be subjective; however, the common goal is to maintain adequate response times by measuring and optimizing an entire Web environment's ability to sustain concurrent users and transactions. Further, running different tests such as benchmarking, scalability, endurance, etc., is the only comprehensive way to accurately predict and optimize end-to-end performance of a Web application.

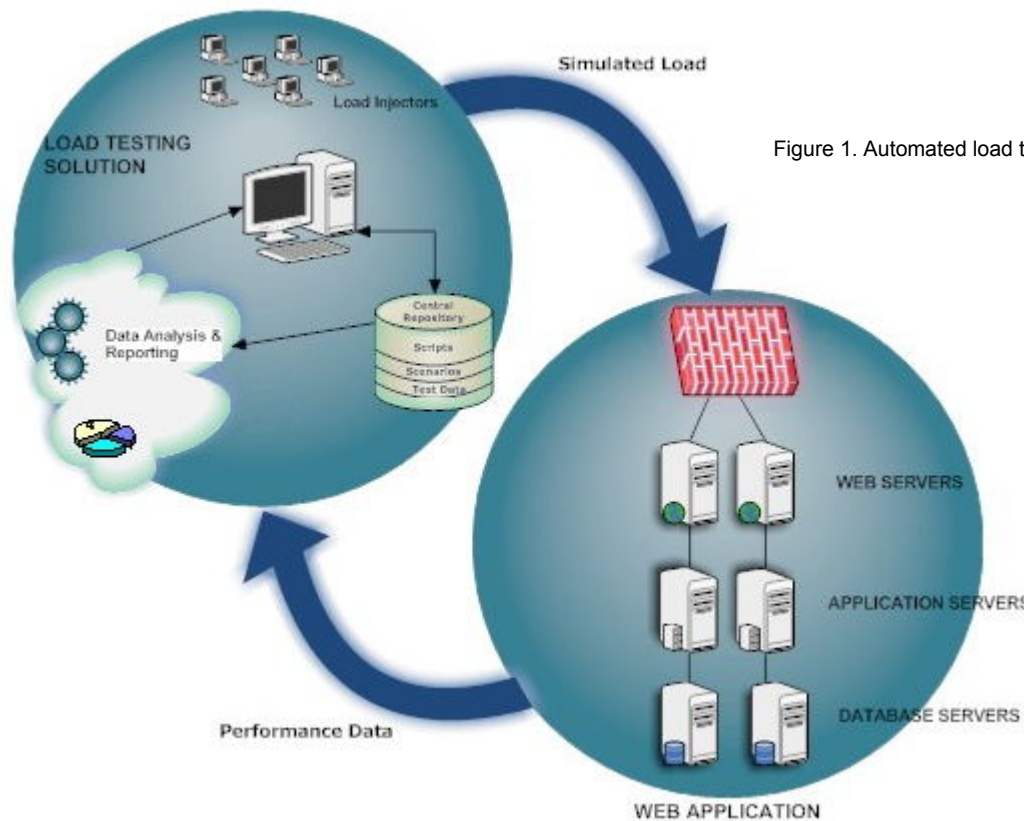


Figure 1. Automated load testing a typical Web architecture.

The most common steps you would take during a typical, automated load test process begin with defining your performance goals and test strategy. This includes determining your user load requirements and identifying the processes and components that need to be tested.

Briefly, the execution of a test typically starts at your Test Center where virtual users and scenarios are created to emulate a real-user environment. Virtual user scripts are recorded by capturing business transactions, and modeled to be replayed by many virtual users. Tasks are defined, scenarios are created, and numbers of virtual users assigned to simulate a real-life user load. You then configure monitoring of the components that you are testing. At this stage, the load injectors (Figure 1) start simulating the virtual users and execute all the tasks and scripts defined in your scenario, and the performance data collection is initiated.

Depending on the size of your Web environment, the results collected from this type of test, as outlined in Figure 1, will typically be a vast amount of data related to the different components of your Web architecture. In order to make sense of all of this data and quickly turn them into easy to understand metrics, your test solution must be equipped with specialized and integrated data analysis tools and reporting features designed specifically for easy data extraction, analysis and reporting. We will discuss these features in more detail later.

Load testing solutions that include these specialized tools can be used to provide answers to many specific types of questions coming from different members of your organization. Whether you are looking for answers about the performance of a currently deployed application, or projections of an application's response to various configuration changes and load patterns, your test solution should provide the actionable answers you need.

In a test environment, a high-end load testing solution should allow you to run many different types of tests, each intended to provide answers to specific performance-related questions about the application:

- ***What load will my application support for a given level of performance?***
- ***How will my application perform in time under nominal load?***
- ***How will my application react to spikes in load levels?***
- ***What changes do I need to make to maintain performances at higher load levels?***
- ***What do I need to do to increase the performance of my application?"***
- ***How does the new version perform as compared to the last one?***
- ***How does the application react to loss of IT resources?***
- ***How will the load be balanced between my servers?***
-

In a production environment, the application is tested under the real life load. The questions are of a different type and the answers more urgent in nature. They typically have to do with understanding the reasons behind poor performance and with finding ways to quickly fix the problem.

- ***What is the end user experience in terms of performance?***
- ***What transactions are below the acceptable performance level?***
- ***What system resources are becoming dangerously close to the acceptable limit?***
- ***What can I do to improve performance immediately?***
-

In order to provide responses to these latter questions regarding an application in production, a testing tool must be usable in "performance debugging" mode. This means that the same tool that is used to test the application under the load of simulated users can also be used to measure its performances in production, and provide the tooling to quickly understand where the issues lie. A testing solution that is able to support both pre-deployment and production environments must have certain characteristics such as real time monitoring, no-load monitoring, and agent-less performance collectors for a zero or near-zero overhead requirement on application resources.

Just the same, in a business context, the questions are no longer about the application itself, but about what the impact its performance characteristics have on current or future business:

- **How much of an investment do I have to make on hardware?**
- **Can I take full advantage of the Christmas shopping season?**
- **Can I commit that my software product will support my customer's requirements?**
- **Am I at risk of losing brand equity or customers to the competition?**
- **Can I support the 25% increase in traffic called for in my business plan?**
-

For a single Web testing solution to bring answers to these questions, it must have the right combination of design characteristics, data analysis tools, and reporting features.

THE RIGHT FEATURE MIX

End-to-End Performance Measurement. This is one of the features that differentiate high-end load testing solutions. Measuring only the Web-application response time, provides valuable information, but does not relate it to the performance of the underlying hardware and software infrastructure. Without that information, it is virtually impossible to understand where the source of performance bottlenecks is or to evaluate the impact of higher load levels.

No-Load Data Collection. Monitors or “collectors” that gather performance statistics from the application back-end perform the data collection process. The centralized activation of these collectors from the test console allows the selection of only those monitors needed for a particular test run. For the test solution to be usable in a performance-debugging mode on an application in production, these monitors must have a zero footprint installation on the application side, and must add negligible overhead on the application as they gather the performance data.

Full Data Integration is essential to allow meaningful cross analysis of data gathered from all of the components of the application. The ability to relate an HTTP request response time to a particular configuration parameter in the application server, overloaded cache, or a poorly performing database query, is an absolute necessity to efficiently “debug” the performance of a complex Web application. All enterprise-class load testing tools support add-on monitoring modules to collect end-to-end application performance data. However, only the test tools with the highest level of integration of these modules will provide the best capability to perform cross analysis and correlation of data generated from all the monitors and the load injectors.

Real-time Monitoring enables monitoring of the entire Web application environment while under a load and visualizing performance data while the test is running. This is especially important for QA Personnel and Developers to get immediate feedback. Even more important, the ability to use the same tool to real-time monitor the Web application while in production (with no virtual users, or only a few, and with minimal overhead) turns your load testing solution into a sophisticated performance debugging tool. The same scripts used in load testing can be used as synthetic transactions for response time monitoring while in production, and all the cross analysis features are available to identify the sources of performance issues (and directly compare the data to that observed in load test runs).

Automated Anomaly Detection helps detect trends or exception conditions that can be the cause of performance issues. This is a specialized, configurable feature that automatically combs through the massive amount of test data, analyzes it, and highlights areas that may be the origin of a loss of performance. These exception conditions should include threshold on the amount of resources used (for example: disk space, network bandwidth, cache use), and trending information useful to detect problems such as memory leaks. More advanced tools will label them by severity level and suggest a course of action to remedy the problem.

Customized Views. Whatever type of test you run, whether benchmarking, load balancing, scalability or any other type of test, the results will provide an endless amount of value if you can quickly and flexibly view them from different angles. Extracting value for a particular user is provided by the selection and analysis of the data significant to that user, and by the appropriate presentation in a format suitable for that person. An efficient integration of data analysis and reporting capabilities provides this ability to leverage the same test results through the creation of different views. See Figure 2 below.

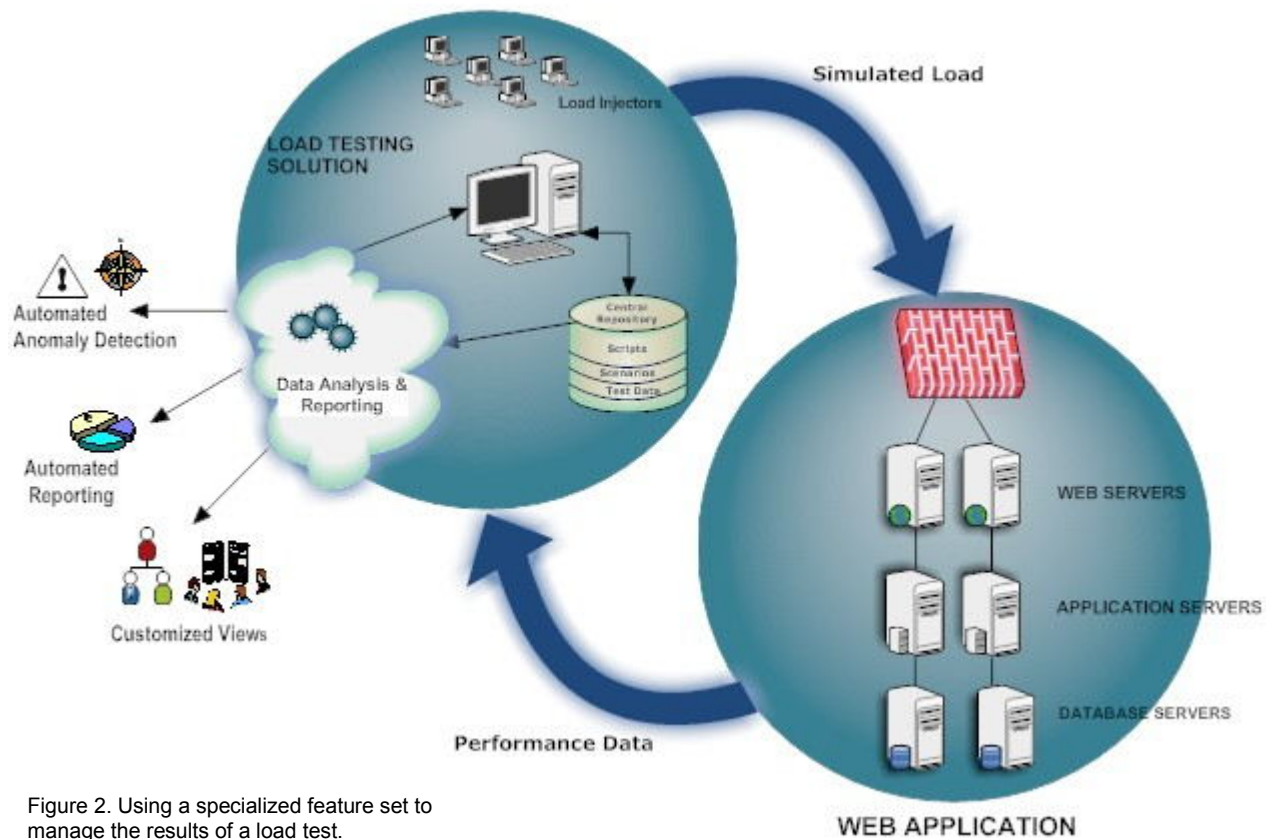


Figure 2. Using a specialized feature set to manage the results of a load test.

Automated Reporting. Beyond an immediate increase in productivity, automated reporting combined with customized anomaly detection, provide the ability to create customizable, reusable views. It also returns a quick, repeatable reading of any test run at the push of a button, for the benefit of a particular user. For

example, a database administrator will want to quickly assess whether a poorly performing query or badly configured parameter is causing a slow page load. A powerful reporting tool will allow you to create and define report templates which can subsequently be used on any test result. (Figure 2). This means that you can drag and drop your template on a test result folder and the appropriate data will instantly extract and automatically populate the template. The data is then saved to a report and ready for distribution. Repeat the same step using results from a different test to receive the same type of report on that data.

LEVERAGING TEST DATA

As discussed above, the appropriate mix of design features and functionalities provide the ability to extract the right data for the right people. This process, even with the appropriate mix, requires skill and time, both in short supply. The key to extracting maximum value out of each test run therefore lies in the ability to leverage past analyses, anomaly detection, and reporting efforts. This implies a kind of “record and playback” feature giving you a “memory” of past analyses and reports in order to replay them automatically on any test result.

For example, the cross analysis of performance data coming from an Oracle database, with observed transaction response times and server resources, can help detect slow performing queries or other database related issues. The capability to replay the same analysis and automatically generate reports from it not only significantly improves productivity, but it ensures consistent analysis and reporting. Applying pre-existing analysis and existing report templates is also especially useful to quickly identify performance issues while in production. In this same scenario, the ability to replay an automated pre-configured anomaly detection scheme will also assist in pinpointing the cause of performance loss.

Another benefit of applying pre-existing analysis and automated reporting on test results is the ability to generate consistent reports for each type of test (benchmarking, stress testing, endurance testing, scalability testing etc.) The reports each focus on a particular aspect of an application’s performance and can then be re-used for different applications. In order to take advantage of such a record and replay feature, it is critical that the recording part of the activity does not create additional work for the Engineer producing the initial data analysis, anomaly detection and reporting. Rather, this should be a by-product of his or her work. The replay part should also be as simple as a drag and drop operation on any test result to quickly generate reports for various people in your organization.

With this ability to easily extract value from the massive amount of test data and produce various types of analysis and reports, different test runs can be compared with consistency and production results can more easily be compared with test runs. Ultimately, the value of your test solution is extended to many more members of your organization.

LOWER COST OF OWNERSHIP – ONE TEST, VALUE FOR EVERYONE

There is a well-documented cost associated with poor Web application testing, but the selection of an enterprise-class testing solution also has many implications both in terms of cost and in terms of return. The licensing and software maintenance costs can vary substantially from one vendor to another, as does the scope of the solution. The value here resides in licensing only for what you really need and for what you will use. The return, however, is more difficult to assess as it involves considerations such as productivity and ease of use, making the total cost of ownership difficult to evaluate. What kind of personnel is required to develop test scripts and test scenarios? Is outside assistance necessary? So, the real question is not only what each solution can do, but what you will actually do with each product. This is not only a function of the solutions offered by vendors, but is more about the match between a particular solution, your application’s characteristics, the personnel involved in using it, and the expectations of various people in your organization.

However, one thing is certain—the more difficult it is to extract value out of each test run; the more costly it will be to find answers to your questions and the less value your test solution will have.

GETTING THE RESULTS YOU NEED WITH QTEST

QTest™ from Quotium Technologies is an automated, enterprise-class load testing and monitoring tool specifically designed to troubleshoot performance and increase the overall efficiency of complex Web-based applications. QTest is perfectly suited for heavy load tests (multiple deployments with well over 10,000 concurrent users).

The Quotium Center has an intuitive Graphical User Interface which ensures real-time availability of all test data and a simple and clear presentation of results. Everything is performed in the Quotium Center—from script editing, test configuration and execution to viewing the results as well and automatically generating reports.

Especially important, all monitors can be set and configured from this single interface, and all performance data gathered is clearly presented in graphical dashboard views. Whether you need general information from the customizable views or want to see infrastructure and application anomalies, or even drill down to the status of database queries, everything is efficiently organized in the Quotium Center for easy viewing. (See Figure 3.)

Beyond the benefits of ease of use, this provides for powerful cross-analysis of all statistical and graphical data. It can then be manipulated, compared and merged to provide comprehensive views of the various resources used for a particular transaction or at a particular time. These cross analysis features are also available in real time, and therefore, in monitoring mode, whether under simulated load or in production. Combined with the availability of monitors for many platforms and third party applications, this integrated design provides a true end-to-end view and analysis of the most complex Web applications. If you are connected to the Internet, you can [click here for more information about QTest](#).

Leveraging this centralized test management and analysis space, two additional QTest features give you the ability to extract maximum

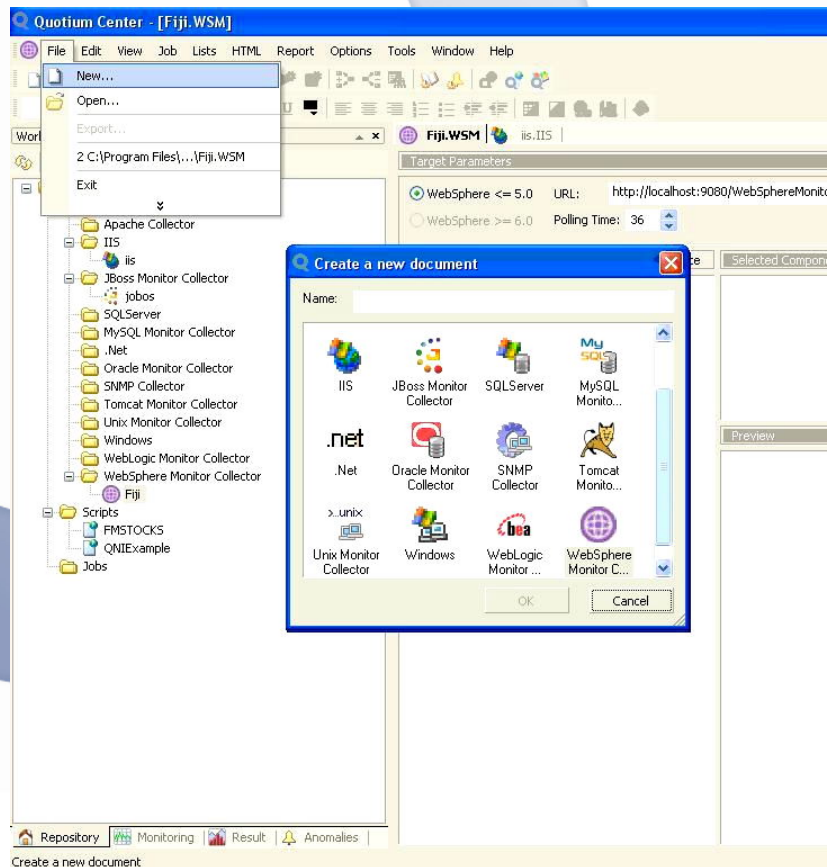


Figure 3. The Quotium Center.

value out of each test run. The **QTest Anomaly Profiler** is used by Engineers and Developers to analyze test results and identify possible performance-related issues of an application. This analysis, performed by skilled personnel, can be saved for future use in different anomaly profiles that can be run against any set of test results to automatically “replay” the analysis with a simple drag and drop operation. The second feature is the **Report Designer**, which provides for the creation of MS Word templates that can be “applied” on any test result to produce repeatable, consistent, professional looking reports, from very high level to very technical documents.

These tools enable you to extract key performance data, in a number of different ways for every level of your IT organization. Even non-specialists can easily create sophisticated reports that include actionable answers for all IT stakeholders.

ANOMALY PROFILER

QTest eliminates the manual process of searching for defects and bottlenecks by doing it for you with its built-in **Anomaly Profiler**. This tool, integrated into the Quotium Center, automatically detects, locates, and traces even the most sensitive transactional, system and application anomalies in your infrastructure. It can be applied to any test result using a simple drag and drop operation. Test Managers can also define their own anomaly profiles to provide different views of the test data and focus in on particular areas.

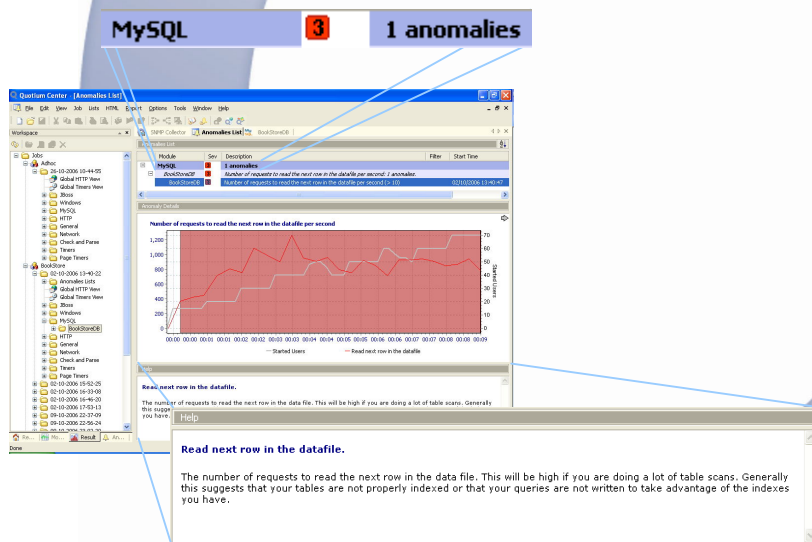


Figure 4. A database problem is detected and problem description provided.

Looking at the example in Figure 4, you can see that QTest has detected one potential problem relating to a database. Problems are labeled by severity and this particular problem is quite severe as it has a rating of 3 (1 being the lowest). The highlighted issue also explains what the problem is and why it is there. This process takes seconds to produce rather than hours that it would take to detect this issue in a manual procedure.

The **Anomaly Profiler** enables you to identify the origin of performance degradation for each component and measure the impact optimization has on the entire infrastructure. It will comb through all of the information generated from the load test, analyze it for you, and highlight the areas from the data that may be the cause of a bottleneck in the system. It will also suggest a course of action to take to remedy the problem. The detector's built-in rules do not require that the person running the test be an expert in the technology they are testing. [Click here for more information.](#)

QTEST REPORT DESIGNER

The QTest Report Designer enables you to create report templates easily and quickly in Microsoft Word format. This powerful, new feature will significantly reduce the time it takes to create customized reports. With the **Report Designer**, after you have created your templates, you can drag and drop them on any test result and automatically populate sophisticated reports with the results data.

Using simple drag and drop operations, graphs, tables, summaries and other selected test data can all be dropped at the appropriate place into a template *once*. (Figure 5). These elements are automatically combined with the existing text, images, and other existing Word compatible content to produce professional looking reports.

Once a template has been defined, it can be reused on any test result. Simply drag and drop the template on a test result folder and *voila*, the appropriate data will be extracted and automatically populate the template and saved to your report, ready for distribution.

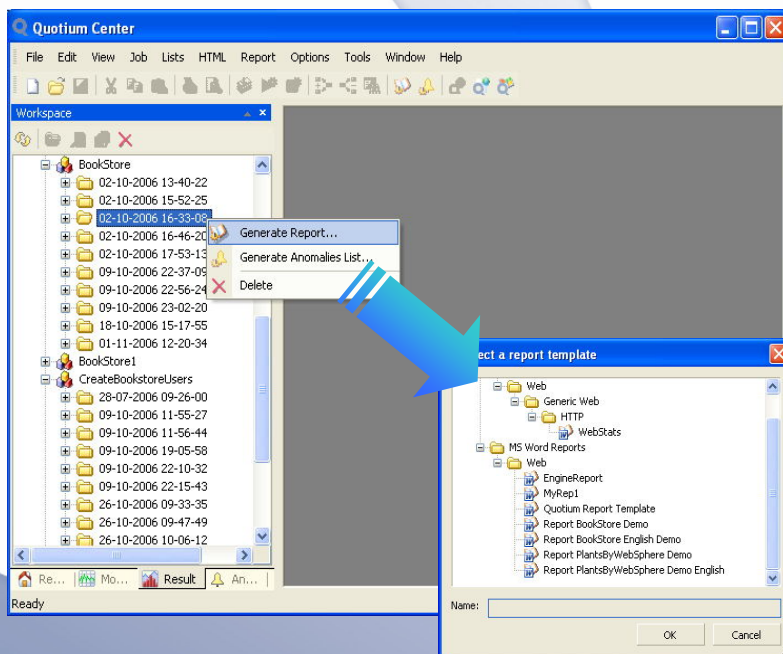


Figure 5. Drag and drop your template on a test result folder and *voila*!

Standard document formats and content such as tables of content, logos, introductions, headers, etc. are repeatable and can be applied to all test reports for consistency and ease of use, test after test. (Figure 6) Reports can be automatically generated in several industry standard formats for insertion of the load test results and performance analysis in wider scoped reports. [Click here to learn more about the Report Designer.](#)

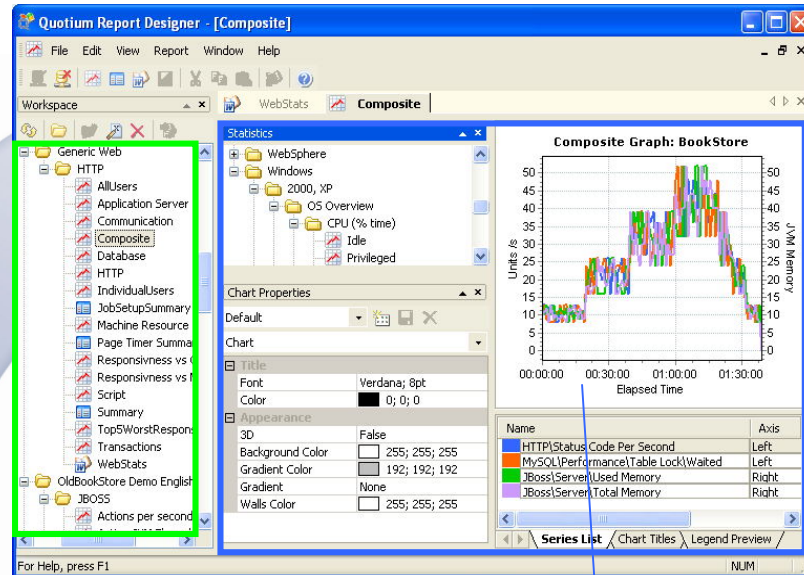


Figure 6. Graphs & tables are saved here for future use.

Graphs & tables built here allow you to customize the content, appearance, statistical operations (running average, percentile and more), scales etc.

CONCLUSION

The need to deploy more complex, faster performing Web applications presents new challenges and requires adopting new practices. As a consequence, significant investments are made in Web testing tools. However, in order to turn your test data into tangible business results, it is essential that specific features be carefully considered during your tool evaluation process. In particular, robust, advanced reporting and anomaly detection features that give a global understanding on the application's response, along with actionable answers are the keys to a significant return on investment. Again, the more difficult it is to extract value, the less value will be extracted and the more costly it will be to extract it.

Quotium's QTest is a sophisticated enterprise-level solution that is affordable. Whether you want to analyze details or see the big picture, QTest can deliver the right data to the right person. In other words, its integrated design, automated reporting and anomaly detection features add superior business intelligence capability to the QTest value proposition. With an advanced testing solution that includes the tools required for operational and strategic decision-making, you can reach your business requirements, quickly and successfully.

ABOUT QUOTIUM TECHNOLOGIES

Quotium Technologies is a global leader in Web application testing and is the value leader in the load testing market. Our enterprise-class products combined with lower costs, equal the best value in the industry. Quotium Technologies' products give medium to large enterprises the Web performance tools they need to increase customer satisfaction and grow brand equity. Quotium's tools identify application and database performance issues, reduce time to deployment and protect a brand's operational excellence. Some of Quotium's clients include Hachette Book Group, Paramount Pictures, Choice Hotels International, Sony Semiconductors, JP Morgan, Kyrocera Mita, HSBC, and Freddie Mac. Quotium has designed its licensing structure to provide the "best fit" pricing to match each individual project requirements and budget. Founded in 2002, Quotium Technologies North American headquarters are located in Wakefield, Massachusetts.

Click on the following links to learn more about QTest or to download the full-version trial:

- [QTEST Datasheet](#)
- [FAQs](#)
- [Download QTest](#)

Contacting Quotium Technologies

Contact us today to set up a personal, live Web demo of QTest!

Quotium Technologies Inc.

201 Edgewater Drive, Suite 280
Wakefield, MA 01880
Tel: +1 781 - 213-9200
Fax: +1 781 - 213-9282
Email: marketing@quotiumtechnologies.com
<http://www.quotiumtechnologies.com>

Quotium Technologies SA

84-88. Boulevard de la Mission
Marchand
92411 Courbevoie, FRANCE
Tel: + 33 01 49 04 70 70
Fax: + 33 01 43 33 92 00
E-mail: info@quotium.com
<http://www.quotium.fr>

Quotium Technologies UK

Golden Cross House, 8. Duncannon
Street
London WC2N 4JF
Tel: +44 (0) 207 748 - 46273
Fax: +44 (0) 207 748 - 45117
E-mail: info@quotium.com
<http://www.quotium.com>