

Borland®

THE OPEN ALM COMPANY

dynaTrace
software

Tracing Performance Issues to Their Source

ACCELERATING ISOLATION AND RESOLUTION OF APPLICATION
PERFORMANCE BOTTLENECKS WITH THE MODERN APPROACH
TO LOAD TESTING AND DEEP DIAGNOSTICS USING
BORLAND® SILKPERFORMER®.

WHITE PAPER
APRIL 2007

Contents

Executive Summary3

Performance and Time-to-Market is a Key Success Factor of Business3

Performance Problems are Difficult to Diagnose3

Reducing Problem Resolution and Project Cycle Times4

Enabling World-Class Performance Testing and Tuning5

The Integrated Diagnosis Process6

Business Benefits9

Executive Summary

Optimal performance is a key requirement for today's mission-critical business applications, which have become more and more complex as trends like Service-Oriented Architectures (SOA) and Enterprise Application Integrations (EAI) become more prevalent.

However, load testing solutions alone have failed to ensure that applications will meet their performance goals in time before deployment. A primary reason is that while load testing tools are useful at identifying performance bottlenecks, they often fail to guide their users in exactly how to resolve them. As a result, time is wasted in what might be called a "blame game" between QA and system engineering about who is responsible to finally fix the issue, which results in excessive cost overruns and missed release schedules.

Only a load testing solution integrated with in-depth diagnostics can really provide what end-users are looking for: a way to quickly detect, isolate and resolve performance bottlenecks in distributed, heterogeneous application environments. Borland® provides just such an answer with SilkPerformer® with dynaTrace Diagnostics®. This application enables clear communication and automation to solving problems by identifying, isolating and helping to resolve performance and stability issues that occur under load.

Performance and Time-to-Market is a Key Success Factor of Business

Most of today's mission-critical business processes are driven by IT applications. The performance of these applications is a key element in today's competitive business environment. Application performance equals business performance – application failures or even poor response times can result in catastrophic costs such as missed revenue, lost customers and brand damage. Thus, validating that business applications correctly fulfill business requirements prior to deployment – from both a functional and a performance standpoint -- has become a central point in IT management.

Competitive advantage is usually dependent on time-to-market and superior user experience. Performance issues can delay or cripple critical business services, resulting in competitors taking market share, users losing productivity and customers abandoning critical tasks. Getting to the root of performance issues as early and as quickly as possible is the key to protecting business continuity and the bottom line benefits of the services that IT delivers.

A difficult, cumbersome process for performance-problem detection, isolation and resolution not only puts your application performance and customer satisfaction at risk, but also significantly lengthens project cycle times and time-to-market – putting the success of your business at risk.

Performance Problems are Difficult to Diagnose

The complexity of today's business-critical applications continues to increase as new technologies are introduced alongside or on top of existing legacy applications. New application development trends like SOA and EAI support this complexity, resulting in highly distributed and heterogeneous multi-tier application architectures that are not only difficult to manage, but also difficult to test and tune for optimal performance.

Additionally, the large number of available frameworks like Java™ EE, .NET, Hibernate® or Struts, allows faster development of complex applications. As business functionalities can only be supported by the interoperations of several systems, messaging infrastructure like messaging services, remoting protocols or Web Services play a central role as well.

While these technological developments often decrease the cost of initial development, the resulting applications or integrated services can become much more complex to test and tune. This can easily result in applications not performing as expected. Efficient performance diagnosis and tuning is then essential for profiting from the savings in reduced development costs.

The important question now is how much time and effort it takes to reach the required performance goals. For today's complex distributed business applications, an efficient performance testing and tuning approach requires close collaboration between all stakeholders involved in the performance optimization process – including QA/Performance Engineers, Application Developers, Database Engineers, Application Architects and System Engineers.

Reducing Problem Resolution and Project Cycle Times

Performance requirements from a business perspective need to be validated and assured at a technical level. From a business level, we can categorize supporting or enabling IT processes with a number of parameters that define performance or quality of service. **Availability** measures the availability of the service to users that consume this service. **Responsiveness** tells us how quickly a service is delivered – and is the key metric end-users care about. **Efficiency** tells us how efficiently the service is delivered (how system resources are used). **Scalability** tells about the ability of the application to cope with a growing number of users and data.

Performance testing verifies the responsiveness and reliability of an application under a certain scale. However, as every application that gets performance tested eventually exhibits performance problems, testing alone is not enough. Naturally, you also want to iron out the performance bottlenecks detected during testing to make the application stronger – that is exactly where performance diagnosis and tuning comes into play.

Performance diagnosis is the analysis of performance issues by incrementally drilling down to the root cause of the problem. In today's application architectures, this process is highly complex and challenging. It requires in-depth knowledge of the inner workings of the application, and the means of tracing of problematic transactions down to a source code level. All these features have to be provided for distributed applications running on various prominent platforms like Java™ and .NET.

Besides support at technical levels, a solution has to support collaboration between all stakeholders on the development teams. Poor communication across departments is frequently a reason for high costs in solving performance issues.

Today, QA often detects performance problems in staging environments and communicates those issues to developers. Developers however cannot use ambiguous information like “*Customer Transaction X is slow*” to detect the root cause of a problem. Developers also may find it difficult to reproduce failures on their development systems. This is because development environments tend to be small single server environments compared to big clustered distributed staging environments. So a lot of effort is spent trying to reproduce the problems found by QA in expensive staging or production environments. This process is shown in Figure 1.

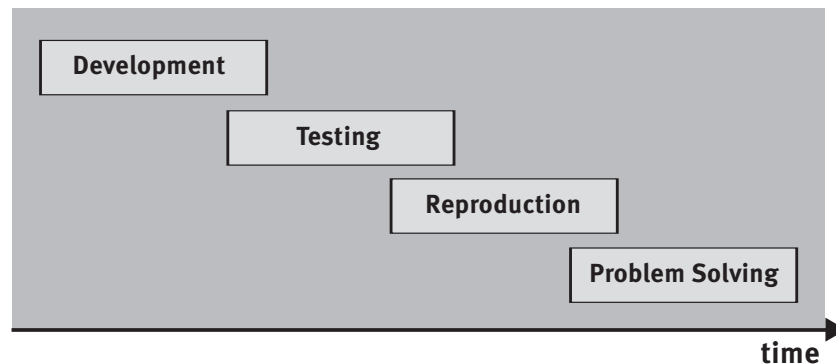


Figure 1 Classical Performance Problem Solving Process

The obvious answer is to directly collect this information during performance testing. This requires a specifically designed solution that allows monitoring of transaction execution paths across heterogeneous distributed application components. In addition, the solution must not modify the behavior of the application under load and must monitor with a minimal performance overhead. Also, the information collected must contain relevant data that developers and system architects can use to reconstruct the error from the monitoring data, without having the need to reproduce it on their system. This saves time and also money, as shown in Figure 2.

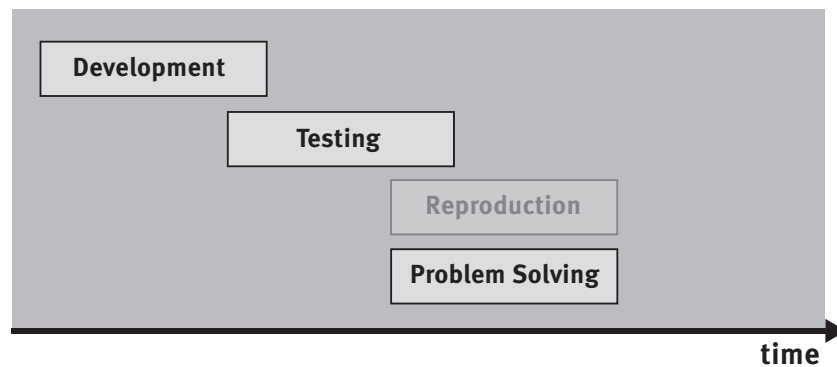


Figure 2 Improved Performance Solving Process

Enabling World-Class Performance Testing and Tuning

Performance testing and tuning must focus on the following key success factors:

- **Accurate emulation of real-users** accessing the target application in order to realistically test its behavior under realistic load conditions.
- **Efficient reporting** of performance test results and detected performance bottlenecks from an end-user perspective.
- **Top-down analysis** of unveiled performance bottlenecks from its symptom (e.g., unacceptable response time) to its root cause even if it is located in the application's code (e.g., chatty application, long running SQL, etc.).

The Integrated Diagnosis Process

Efficient performance diagnosis is more than tools. It requires an integrated solution that enables clear communication and automation to solving performance problems by aligning developers, performance engineers, and system and application architects.

Borland SilkPerformer with dynaTrace Diagnostics provides an integrated solution that follows this principle. Performance engineers use Borland SilkPerformer to define end user transactions and then use them to run load tests against the application to identify performance problems. At the same time, the application is completely monitored down to the component/method/code-level using dynaTrace Diagnostics at nearly no performance overhead.

Then, application performance metrics such as end-to-end response time, time spent in various application components like the J2EE stack or database actions are correlated with the current load on the server. This information can be monitored in real-time during a load test using the Performance Explorer in SilkPerformer as shown in Figure 3. This provides a comprehensive overview of the behavior of the application. Silk Performer's Performance Explorer answers questions like:

- What are the top consuming components in the application?
- Is the application slow or is there a networking problem?
- What is the effect of user behavior on memory management?

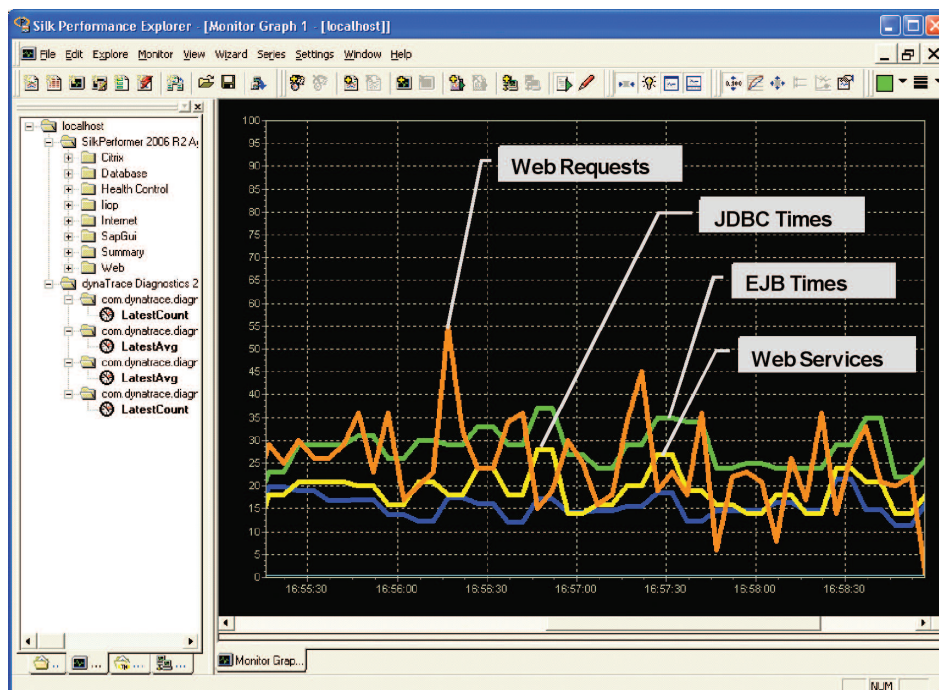


Figure 3 Correlate Web Page Times with In-Depth Application Performance in Realtime

After the load test is finished, the load testing report provides a clear entry point for deeper analysis of performance problems. With the integrated solution, SilkPerformer and dynaTrace Diagnostics enable a seamless interface to drill down to the exact location of the performance bottleneck, whether it's an excessive response time, overloaded CPU or memory or other critical metric in specific software components like Servlets, EJB™ or JDBC™ code. As shown in Figure 4 further diagnosis using contextual information (e.g., memory allocations, synchronization issues, etc.) is utilized to pinpoint the exact reason why problems occur, thus identifying its root cause.

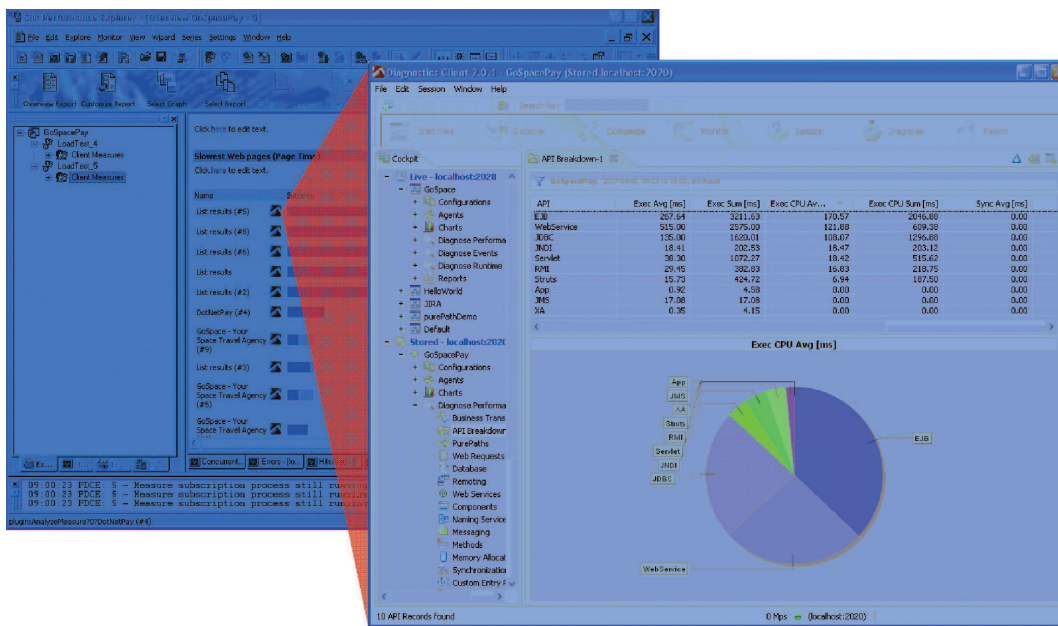


Figure 4 Drill down from load test results to detailed application breakdown

In cases where problems are related to specific requests (for example, a missed service level or erroneous response of a particular transaction) detailed information is required related to the processing of this single request. The actual path the individual request has traversed through the system can be reconstructed by using dynaTrace PurePath® technology as shown in Figure 5. dynaTrace Diagnostics records each single execution path and includes context information captured during execution of the load test. Unlike other tools, there is nearly no performance overhead introduced by dynaTrace that could modify the behavior of the application, consequently affecting the accuracy of the diagnostics.

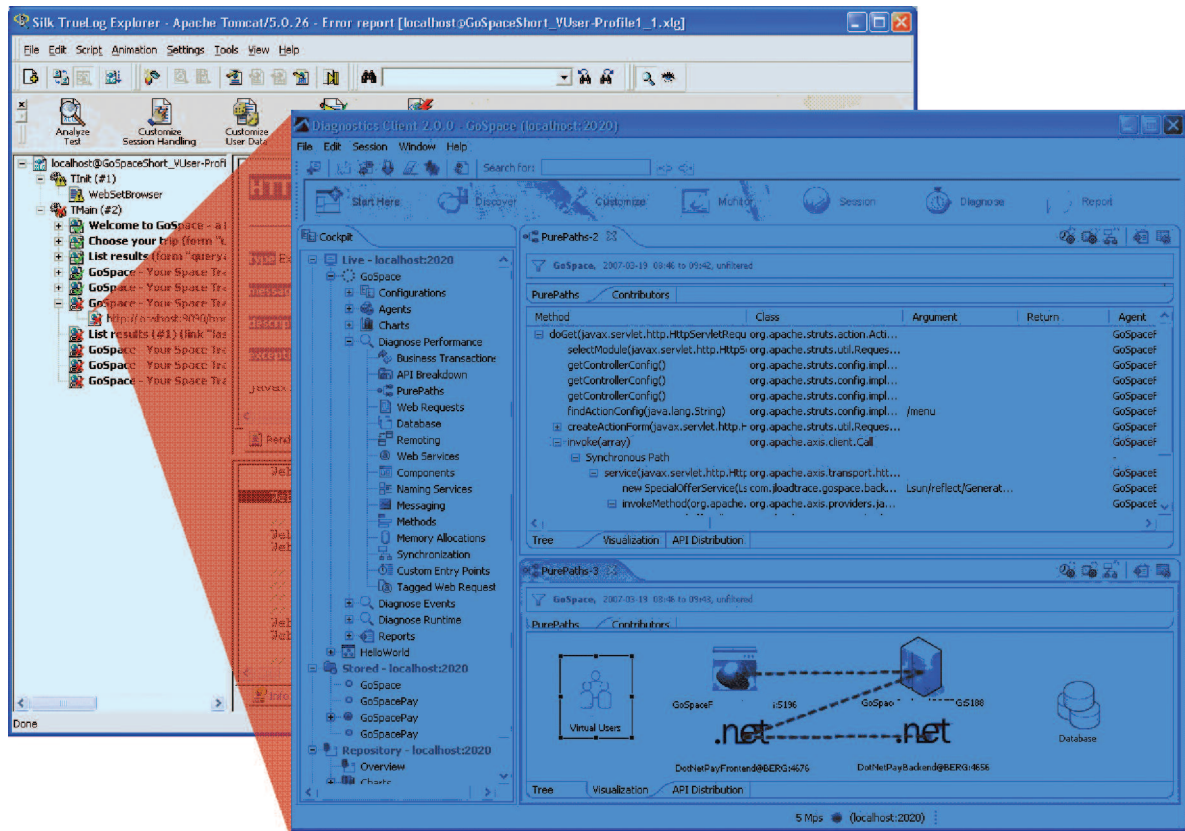


Figure 5 - Drill down from a single web transaction to distributed heterogeneous PurePath

Other solutions often only provide aggregated results that group all transaction information together making it very difficult to isolate performance issues related to specific transactions. This is a core strength of the combined SilkPerformer/dynaTrace Diagnostics approach. QA and development can reconstruct each specific transaction that caused problems without having to guess what specific transactions were doing within a sea of aggregated values. Development teams are provided what they need to analyze the collected information offline to directly investigate problems down to code level. Using PurePath technology, it is possible to identify problems including memory leaks, synchronization issues, inadequately configured frameworks and excessive remoting/messaging in the context of a single transaction across servers in heterogeneous .NET and Java environments.

Business Benefits

The integrated Borland SilkPerformer and dynaTrace Diagnostics solution provides significant benefits to your software delivery process because it improves the communication between Development and QA to quickly resolve performance and stability problems without dispute or delay.

Testers and performance engineers can easily document the root cause of performance and stability issues down to the offending line of code, even in complex, highly distributed applications. They do not need any programming knowledge or access to source code to deliver an offline diagnosis report that allows development to efficiently resolve issues.

Development practitioners such as developers, architects and database engineers become more productive resolving performance and stability issues that occur under load. The time intensive, tedious, error prone and often impossible process of manually correlating log files from different back-end systems with the client-side load testing results becomes obsolete, which significantly accelerates problem resolution times. Additionally, developers can now easily configure the in-depth monitoring information for their application that should be collected during a load test. These configurations are then automatically applied to the testing environment, which also makes the time consuming error reproduction process obsolete.

What is the bottom line? SilkPerformer and dynaTrace Diagnostics deliver measurable value by:

- Ensuring application performance before deployment, reducing often catastrophic, immensely costly application downtime in production.
- Reducing the time required by development teams to isolate performance issues.
- Reducing the time spent in meetings or conference calls determining who actually owns the problem.
- Accelerating the resolution times of performance and stability issues by providing accurate, relevant diagnostics information down to the offending code .

Borland SilkPerformer and dynaTrace Diagnostics, working seamlessly as a purpose-build solution, significantly improves the process of application performance issue isolation and resolution, delivering on the promise made by development teams to deliver fully validated, high-performing business services through IT.

To learn more please visit www.borland.com or www.dynatrace.com.

Borland is the leading vendor of Open Application Lifecycle Management (ALM) solutions - open to customers' processes, tools and platforms – providing the flexibility to manage, measure and improve the software delivery process.