



Upside Research

J2EE Service Management: Are You Ready?

Authored by:

David Kelly and Heather Ashton

*Upside Research, Inc.
www.upsideresearch.com*

CONTENTS

□ **INTRODUCTION**

Java is increasingly being used in business-critical applications, primarily because it offers several advantages over older technologies. However, while most organizations have logically been focused on the task of developing Java applications, more organizations are now beginning to realize that there's another important consideration to creating Java applications—deployment and management.

□ **THE ISSUE: JAVA PERFORMANCE**

Java has enabled organizations to take application development to a new level of functionality. While the increase of Java in the enterprise environment solves many of the previous limits of application development, it does not eliminate the traditional concerns of enterprise applications, namely the application service management challenges.

□ **HOW TO BEGIN ENSURING JAVA SERVICE NOW**

IT managers should make a number of service considerations when delivering J2EE applications to the organization. Identifying these performance considerations and beginning to anticipate these issues during development is the first step in being able to better manage J2EE applications.

□ **VANTAGE: WHAT A GOOD SOLUTION FOR MANAGING ENTERPRISE JAVA APPLICATIONS SHOULD PROVIDE**

Compuware's Vantage is designed to provide IT operations staff with the highest level of monitoring and analysis capabilities to keep their J2EE applications running effectively.

□ **CONCLUSION**

Upside Research recommends that IT operations managers look for a solution that will not only enable them to respond quickly to J2EE application performance issues, but also proactively manage J2EE applications in a way that meets service levels and contributes to business goals. Compuware's Vantage is a solution that accomplishes both of these objectives.

INTRODUCTION

Most companies are devoting more and more attention and effort to their Java development projects, but what about the flip side—the deployment and management of completed J2EE applications? Unless organizations give some thought and planning into the operations and applications management aspect of Java applications, organizations are increasing the risk of potential problems and failure to meet business needs.

As the technology has evolved, Java is increasingly being used in business-critical applications, primarily because it offers several advantages over older technologies. However, while most organizations have logically been focused on the task of developing Java applications, more organizations are beginning to realize that there's another important consideration (beyond development) to creating Java applications—deployment. To ensure success and meet business goals, organizations developing Java applications need to also consider the management of Java applications and services once they are deployed.

Although the benefits of J2EE are apparent to most IT managers, it is also important to understand that J2EE brings with it a set of new challenges that IT organizations must manage—typically at the same time they're continuing to manage all their existing issues and ongoing development challenges. J2EE is an extremely complex technology, and because of its relative newness, many organizations have even more limited resources than usual to address development roadblocks associated with J2EE. For example, IT operations managers are often faced with a lack of experience among their staff for troubleshooting J2EE applications. Because of the absence of internal expertise for J2EE, developers may be called into reacting to an emergency, making for a very costly fix to the problem.

Instead of being at the mercy of a defective J2EE application or poor infrastructure, managers need a solution that enables them to understand service impact of the performance issues of their Java applications and effectively manage their J2EE services. Specifically, they need to consider both the issues related to developing in a J2EE environment as well as the specific issues associated with managing J2EE services effectively and ensuring that business needs are met.

Compuware Vantage provides a practical approach for IT operations staffs that are concerned about the issues of developing and deploying J2EE services effectively and efficiently. Vantage provides a number of benefits that help IT organizations ensure adequate end-to-end performance and deliver on targeted business goals. This paper addresses some of the challenges and concerns that IT operations managers, development managers, and IT staff may have with their current J2EE environment. It offers solutions for effectively managing IT environments with J2EE in a way that addresses specific business-related goals for application performance. IT managers can use this paper to better understand the issues surrounding J2EE development and management as well as how a solution like Vantage can effectively improve performance and deliver consistent application service to meet business needs.

THE ISSUE: JAVA PERFORMANCE

Java has enabled organizations to take application development to a new level of functionality. Among the benefits that J2EE provides is a level of flexibility and platform independence not previously available, which is important given the necessity for businesses to be extremely agile in a changing competitive landscape. J2EE's template

architectures and ability to recombine components enables IT organizations to meet business demands for applications that are more responsive to current (and even fluid) business situations. In response, IT organizations have added Java developers, as well as Java-based tools and infrastructure to incorporate some of this flexibility and agility to their development efforts. The results have been that Java is now a key building block of many enterprise IT infrastructures.

Although the increased amount of Java in the enterprise environment solves many of the previous limits of application development, it does not eliminate the traditional concerns of enterprise applications, namely the performance and management challenges. And, unlike older technologies that have had the time to mature and enabled internal management expertise to develop, Java and J2EE are relatively new and as a result many organizations lack the in-house skill sets for managing these challenges.

Faced with these issues and realizing that quality can have a significant impact on the business, IT managers must find ways to improve the quality of their released applications through a program that assesses and improves the testing processes that are controlling application quality.

COMMON CHALLENGES WITH DEPLOYED J2EE APPLICATIONS

J2EE applications bring with them a number of management challenges that can overshadow the benefits of the technology to end users and IT operations staff. Because of the newness of the technology, there is a general lack of industry-wide experience in managing deployed J2EE applications. This is extremely challenging for an organization that has invested in either training or bringing in new resources to develop applications in J2EE, only to realize that once the applications are deployed, they bring a whole new set of issues that the existing IT staff is not equipped to handle. Among the other challenges that are frequently experienced with J2EE applications once they have been deployed are:

- Identifying performance issues that may be caused by J2EE's powerful architecture due to the "black box" challenge
- Understanding infrastructure dependencies and their effects on performance
- Identifying and fixing memory leaks that lead to costly downtime

In addition, IT operations staff face the challenge of delivering consistent service levels to the business with J2EE applications, and incorporating their management of J2EE into their overall service-level contracts with the business. Let's take a closer look at some of these key issues:

- **Performance issues related to Java's "black box"** - One of the most frustrating aspects of J2EE applications from an IT operations perspective is the fact that there is no visibility into how the application code is executing. This makes it extremely difficult to pinpoint performance issues because there is no way to follow the execution paths of the code. For example, when a J2EE application has a problem, IT might not know what resources are used to execute methods or how many times a method is being invoked. As a result of not fully understanding what's happening in the application, a developer might spend significant time analyzing methods that are not even related to the offending method or the root cause. Often, when a problem related to performance arises, different areas are suspected of causing the problem, and each representative group (networks, systems, applications) has its own idea about solving the problem. This makes it difficult to create and execute a plan of corrective action.

If there is a J2EE performance expert in-house, many times the only way to get to the root of the problem is to perform a manual tracing to observe the behavior at the Java code level. This method is costly as well as extremely slow, and it fails to meet business needs for timely fixes to performance glitches. In addition, it does not add any value to the business, because often it is a one-time fix that cannot be used strategically for future performance problems.

- **Infrastructure dependencies' effects on performance** - Because a J2EE application is complex in its composition, it may involve a distributed web-based, front-end component that integrates with information stored in a mainframe or legacy system, database system, or multiple systems that reside within the corporate IT infrastructure. The new J2EE application utilizes resources in the infrastructure in ways that hadn't necessarily been planned for during development, and when the application is in production it can execute resource-intensive business transactions that consume high amounts of computing resources. This results in a situation where traditional applications have slower response times or become unavailable for lengths of time, which is not acceptable to business needs. Again, multiple groups may be called in to solve the problem, such as mainframe, database, or applications—each with their own idea of what the problem is. Many IT departments respond by adding computing resources—such as application servers or more processors, memory, or Java Virtual Machines. Unfortunately, this does not solve the inherent problem of an inefficient and poorly scalable J2EE application, and instead drives up IT costs.
- **Memory leaks** - Memory leaks are common in J2EE applications and are another source of performance problems or potential application downtime. In fact, some organizations report routinely recycling their JVM as frequently as every two hours simply to reset memory lost from undetected memory leaks. Memory leaks are caused by programs that allocate Java objects but do not properly de-allocate them. As a result, the available memory in a J2EE application eventually runs out, causing the Java Virtual Machine (JVM) to become unstable. This can lead to application downtime.

The most common way that IT departments try to fix the memory leak problem is by recycling the JVMs. While this does help to clear up the problem, it also necessitates application downtime, which is unacceptable for business-critical applications.

Another way that IT operations may attempt to solve the memory leak problem is by walking through the Java code, which is a slow process and may not necessarily identify the leak because there can be thousands of objects allocated and de-allocated in a given period of time.

They may also attempt to write Java tracking code to match allocations with de-allocations, but again the process is extremely time-consuming and fails to meet pressing business needs.

- **Delivering consistent service levels in line with overall IT service plans** - While the previous three challenges of J2EE production environments focus on specific aspects of J2EE performance and functionality, this last challenge encompasses the bigger picture as it relates to overall IT performance. Most IT organizations have entered into certain service agreements with their business counterparts, ensuring that the applications they provide and support will perform

at certain levels that are necessary for business success. Adding J2EE applications into the mix often presents a unique challenge to IT managers, because they typically have minimal experience delivering service based on J2EE applications. The lack of familiarity often hinders IT organizations in their ability to establish and meet service level standards for the new applications. This often detracts from overall service performance and is difficult to improve under existing conditions.

Since many of the problems surrounding the deployment of J2EE applications consume IT departments, there is little time to focus on proactively meeting service levels. Instead, these strapped resources are too busy reacting in constant fire-fighting mode to the performance, scalability, and memory leak issues that J2EE applications present. As a result, it is nearly impossible for IT groups to be proactive in their approach to effectively manage J2EE applications.

THE BUSINESS IMPLICATIONS OF POORLY MANAGED J2EE APPLICATIONS

The fact that IT groups have very little opportunity to proactively manage J2EE applications illustrates the constant defensive mode that IT operations has been in with J2EE deployment. This constant reactive mode can cause a negative impact on the business, financial, and technical operations of an organization. Among the implications from a business perspective are the following considerations:

- **High costs associated with “band-aid” fixes.** Because most IT staff lack experience managing J2EE applications, there is often a tendency to throw more resources at the problem, whether they be additional development time or infrastructure purchases. This reactive mode can be financially costly, and often it only masks the deeper problems that exist within the J2EE application, rather than providing effective solutions.
- **Application performance problems and downtime.** The instability that can result from J2EE applications deployed without optimization can lead to costly performance issues and potential application downtime. Both of these situations can be damaging from a business perspective, especially if the J2EE applications, or the applications affected by J2EE performance, are business-critical. As a result, deploying an inefficient J2EE application can be detrimental to the business in ways that were never anticipated during development.
- **IT resources are taxed with J2EE problems.** Many organizations focus all of their efforts on leveraging J2EE developers to build and develop complex applications, forgetting that someone has to manage these applications once they are deployed. Without dedicated resources for J2EE applications management or the ability to pinpoint exactly where problems are occurring in a distributed computing environment, IT operations managers must pull resources from other areas, whether they be applications, network, or systems staff, to attempt to solve J2EE problems. This causes a domino effect, resulting in failure to meet other important business needs that may exist in other application-related areas. And, ultimately, it may lead to IT’s failure to meet its overall service levels promised to the business.

Building Java applications is difficult enough for many IT organizations, but it is important to realize that the challenge does not stop with deployment, but rather opens up an entire new set of challenges and problems that must be addressed. The costs to a business are numerous and can be high if a J2EE application is not functioning correctly

once deployed. That is why it is critical for organizations to consider both development and deployment issues when creating J2EE applications. To accomplish this, most organizations need new processes and tools to properly identify, manage, react to, and predict overall J2EE application performance issues, ultimately ensuring the quality of the entire application environment.

HOW TO BEGIN ENSURING JAVA SERVICE NOW

IT managers should make a number of performance considerations when delivering J2EE applications to the organization. Identifying these performance considerations and beginning to anticipate these issues during development is the first step in better managing J2EE applications. Understanding how operations staffs need to be ready for the J2EE applications once they are passed over the development wall will enable organizations to more effectively ensure that J2EE applications perform at acceptable levels.

BEST PRACTICE CONSIDERATIONS FOR EFFECTIVE J2EE APPLICATIONS DEPLOYMENT

The primary goal of the development effort for a new J2EE application is to meet the business needs by providing the level of functionality necessary to accomplish the primary goals of the application. Rapid development is a component of meeting business needs, because often J2EE applications are critical to business operations and therefore have a higher level of visibility than other projects. As such, developers are under pressure to deliver all of the functionality in the application in a compressed project timeline, and many give little thought to how the application will perform in the greater context of the enterprise IT infrastructure once it is deployed.

Upside Research recommends that IT organizations raise the following issues during the development phase of J2EE applications in an effort to offset the most common performance challenges that can result from poorly planned applications:

- **Identify and understand infrastructure dependencies**
 - What existing systems, applications, and databases will this new application connect with to perform its tasks?
 - What are the current load limit levels of these systems?
 - What level of demand will the new application place on them?
Can it be anticipated when it will peak? Is it constant or sporadic?
- **Identify potential memory leak areas**
 - What are the Java object allocation patterns of this new application?
 - Can you anticipate how they will impact performance?
 - Are there ways to create scenarios regarding this in the testing phase of development?
- **Establish a basis for realistic service levels**
 - What is the minimal service level that must be met with this application once it is released into deployment?
 - Will this application have an impact on other existing service level agreements that need to be considered now?
 - Predict or anticipate affects of production conditions
 - Response time budgeting for each component of a transaction
 - Resources required for new transactions

By starting a dialogue during development for some of the issues mentioned above, IT operations staff can gain some insight into the properties of the J2EE applications that they need to manage upon deployment. This can help developers better understand the effect their development efforts can have on the ultimate performance of the J2EE applications, and enable better planning from a management perspective. The next area for IT operations staff to focus on is creating a system for identifying and eventually anticipating potential performance issues with J2EE applications.

BEST PRACTICE CONSIDERATIONS FOR J2EE APPLICATION MANAGEMENT

When J2EE applications are newly deployed in an IT environment by an IT team that has little or no previous experience with them, there may be immediate evidence that the applications are experiencing performance issues. The most important way to ensure success in managing these new applications is to establish systems for identifying and measuring the performance of the applications and the systems that are impacted by the J2EE applications. An effective management solution should include the following ways for operations staff to become aware of J2EE application performance and its effects on the rest of the IT infrastructure:

- **Visibility into production J2EE performance.** While this is an obvious requirement for more effective J2EE management, the important aspect of this capability is to enable operations staff to view the cumulative activity of the J2EE environment without having any impact on the actual application's performance.
- **Minimize the need for J2EE knowledge.** Finding a way to deliver performance details for J2EE applications without requiring the operations staff to be Java experts is an important consideration in successful application management. IT should provide a tool or system that enables staff to view problems and issues in the J2EE applications without requiring a high level of Java expertise.
- **Pinpoint the problem.** Because it is difficult and costly to solve problems without understanding exactly where they reside in the J2EE environment, it is important for IT to create a management environment that pinpoints new problems as closely as possible to the related application or system area. There should be a high-level view that allows operations staff to immediately focus on the most serious problems as well as drill down capabilities to speed problem resolution by getting the right diagnostic information to the experts in the organization quickly.
- **Analysis capabilities.** If the management system for J2EE applications includes a deep analysis functionality that provides insight into potential problem areas, it will enable the operations staff to reduce the risk of performance, stability, and scalability issues. Enabling this functionality in the staging and testing environment will enable developers to simulate potential problems and fix them, thus increasing success levels upon deployment.

Compuware Vantage is one solution available today that can enable organizations to more effectively deploy and manage their J2EE applications by delivering these types of capabilities and enabling IT organizations to address these issues, meeting both IT management and business-related needs for effective applications management. The next section takes a closer look at Vantage and explores how it can help IT operations staff feel confident about their J2EE applications.

VANTAGE: WHAT A GOOD SOLUTION FOR MANAGING ENTERPRISE JAVA APPLICATIONS SHOULD PROVIDE

Compuware's Vantage is designed with the intention of providing IT operations staff with the highest level of monitoring and analysis capabilities to keep their J2EE applications running effectively. Vantage accounts for the often limited scope of Java experience in IT operations staff, and as a result provides a management and monitoring solution that provides actionable insight without requiring J2EE expertise. The solution provides insight into exactly what areas of a J2EE application are causing a problem, including visibility into performance bottlenecks, memory leaks, and scalability problems. Vantage is a comprehensive application service management solution that can enable IT organizations to effectively deliver end-user focused service levels for J2EE applications in the same manner they do other enterprise applications.

A CLOSER LOOK AT VANTAGE

Compuware Vantage enables IT organizations to manage application performance from the end-user perspective. Vantage provides management of applications at the business, transaction, and infrastructure levels. It provides response time metrics as well as end-to-end performance analysis, which enable IT managers to proactively identify and resolve performance problems. Vantage also helps assess the impact of application, workload and infrastructure changes prior to deployment, providing pre-production service assurance.

Vantage provides specific support for J2EE production environments, and includes a number of features and functionality that directly relate to the unique management issues that J2EE applications bring with them into production. Vantage provides service dashboard interfaces for IT operations staff to easily monitor and manage J2EE application performance, identifying any problem spots quickly. The solution focuses on monitoring the complete Java service, including network, server, application monitoring, or even components monitored by other tools. This is important because of the many interdependencies that J2EE applications have with existing IT infrastructure. It provides real-time visibility into performance, and enables monitoring and information capture to be automated, easing the burden on IT.

VANTAGE IDENTIFIES MOST COMMON J2EE PERFORMANCE PROBLEMS

As mentioned earlier, J2EE applications face several specific common performance problems. Vantage is designed to address these issues, enabling IT operations staff to more effectively manage the applications:

- **“Black-box” performance issues.** IT operations staff quickly discovers that J2EE applications in production are a “black box” in terms of execution paths and the lack of visibility into their performance behavior. Vantage provides a comprehensive performance monitor that collects widespread information about J2EE applications performance with a minimal impact on the production application. This provides a high level of confidence in pinpointing the specific problem, rather than wasting time and efforts on incorrect assumptions. When a problem is identified, Vantage can capture deep dive execution trace paths without needing to bring the application down, an important capability for business-critical J2EE applications. In addition, Vantage only collects information at the application level, which focuses operations staff on identifying the problems faster, rather than having to wade through hundreds of hidden system calls to get to the root.

- **Infrastructure dependencies.** J2EE applications can often be inefficient in their use of other IT system resources, including other applications, databases, and systems. Vantage aggregates all application server performance, automatically filtering it into various classes to determine which methods are using the most resources and taking the most time. It also calls out which areas can be tuned for better application scalability—a feature that promotes proactive application management of J2EE applications. If there are certain slow-spots, Vantage can identify them, and hand off specific areas of code for performance tuning. In this way, it effectively helps to optimize the impact that J2EE applications have on the entire IT infrastructure.
- **Memory leaks.** Memory leaks caused by programs that allocate Java objects but do not properly de-allocate them are a source of potential application downtime for J2EE applications. Vantage can help IT operations staff trace the usage of memory by a J2EE application and identify places where programs are allocating Java objects but failing to de-allocate them. Through this tracing functionality, Vantage provides a view into exactly where memory leaks exist. This enables the leaks to be identified immediately, and operations staff to fix the leaks, minimizing the potential for costly downtime and further stabilizing the J2EE applications.

BENEFITS OF VANTAGE IN A J2EE ENVIRONMENT

Vantage provides a number of benefits for IT operations staff, including technical and business benefits. One of the greatest benefits of the solution is its ability to turn an IT operations team from constant reactive mode into proactively identifying and solving problems related to J2EE application performance. By enabling teams to stay on top of J2EE application problems, IT can more effectively deliver on promised service levels, including with J2EE applications, which delivers important business advantages.

Among the other technical and business benefits that Vantage provides are:

- **Fixing the root problem reduces overhead.** By zeroing in on the J2EE performance problems immediately, IT staff is able to reduce the costs associated with searching for hard-to-find issues. They also avoid the risks associated with throwing more resources at a problem, which only provides a temporary fix and drives costs up. Vantage enables IT operations teams to act strategically, keeping management costs low and delivering the most effective management of J2EE applications.
- **Application reliability is improved.** One of the business benefits of utilizing Vantage to discover and begin repairing J2EE application instabilities caused by memory leaks or infrastructure dependencies is the reduction in overall application downtime. Since many J2EE applications are business-critical, any downtime can be damaging to business, and preventing that downtime provides a significant business benefit to the organization.
- **Streamlined management saves IT resources.** With an application service management solution like Vantage, IT organizations no longer find themselves in the difficult position of wasting precious IT resources to fix difficult J2EE and deployment runtime problems. They are able to quickly identify the J2EE problem, apply a specific resource to fix it, and enable the rest of the IT staff to continue performing their value-added functions. Overall, this enables IT to successfully meet other important business needs on time and on budget, while delivering more stable J2EE applications to the enterprise.

CONCLUSION

Java provides an exciting technology for enterprises in their attempts to remain competitive and flexible. J2EE applications have proliferated in recent years as development teams become more familiar with the technology and the tools to develop Java applications mature. The greatest challenge for J2EE applications has become managing them once they are released into production. Because many IT organizations lack expertise in J2EE applications management, this endeavor can be an uphill battle without the proper tools.

Upside Research recommends that IT operations managers look for a solution that will not only enable them to respond quickly to J2EE application performance issues, but also proactively manage J2EE applications in a way that meets service levels and contributes to business goals. Compuware's Vantage for enterprise Java environments is a solution that accomplishes both of these objectives. With Vantage, IT operations staff can effectively manage J2EE applications in the context of a high-performance, enterprise-computing environment. Vantage allows IT managers to analyze performance, prioritize problems, and efficiently solve them according to the impact they have on the business. The net result is a more efficient IT operation that successfully incorporates leading-edge technologies such as J2EE into business-critical systems.

About Upside Research, Inc.

Upside Research is a research and consulting firm focused on helping clients put application development, Web services, business process management, integration, and enterprise infrastructure challenges in perspective. Upside Research helps organizations find practical ways to achieve their IT goals and profit from the diversity of a changing technology landscape.

Upside Research, Inc.
www.upsideresearch.com
info@upsideresearch.com