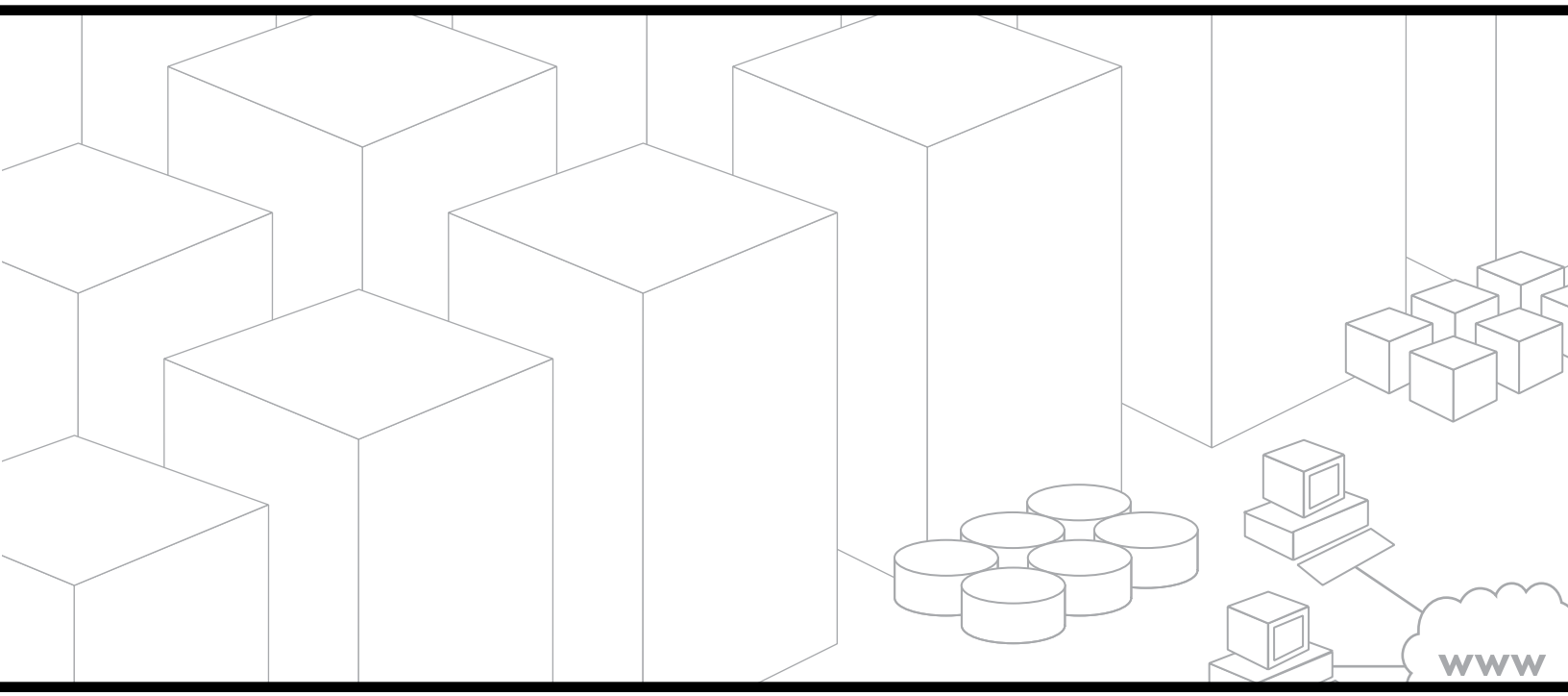
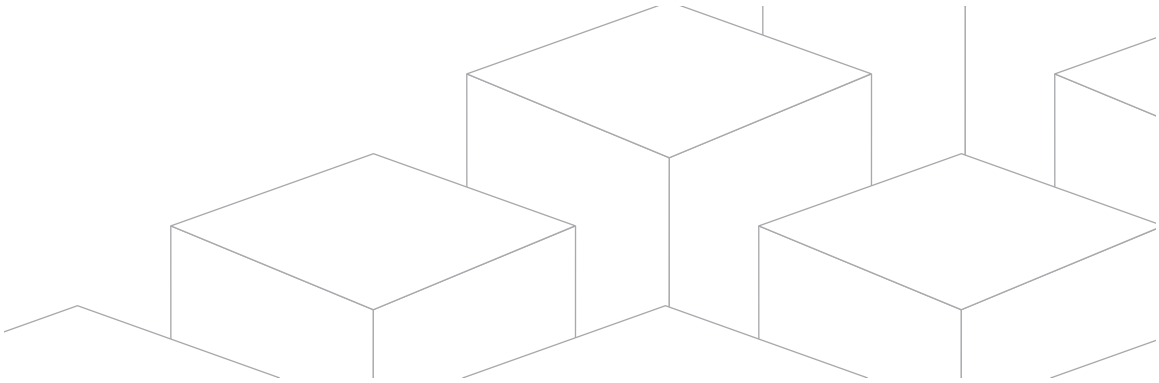
The Role of the Architect in Java™ Application Management



wily technology

Wily Technology, Inc.
8000 Marina Boulevard, Suite 700
Brisbane, CA 94005
U.S.A.

1 800 GET WILY <U.S. toll free>
+1 415 562 2000 <direct>
+1 415 562 2100 <fax>
info@wilytech.com



Abstract

This paper examines the role of the enterprise IT architect in application management in general, and in the management of Java-based Web applications in particular. It considers several alternative definitions of the architect's role within the organization and formulates a generalized characterization of that role. It explores the types of negative influences that can prevent architects from maximizing their contribution to the enterprise, and proposes certain measures that the architect can undertake in order to help neutralize such negative influences-particularly within the sphere of application management. Finally, it focuses on the management of Java-based Web applications and identifies specific technologies and tools that the IT architect can leverage in order to better achieve the business objectives with which he or she is charged.

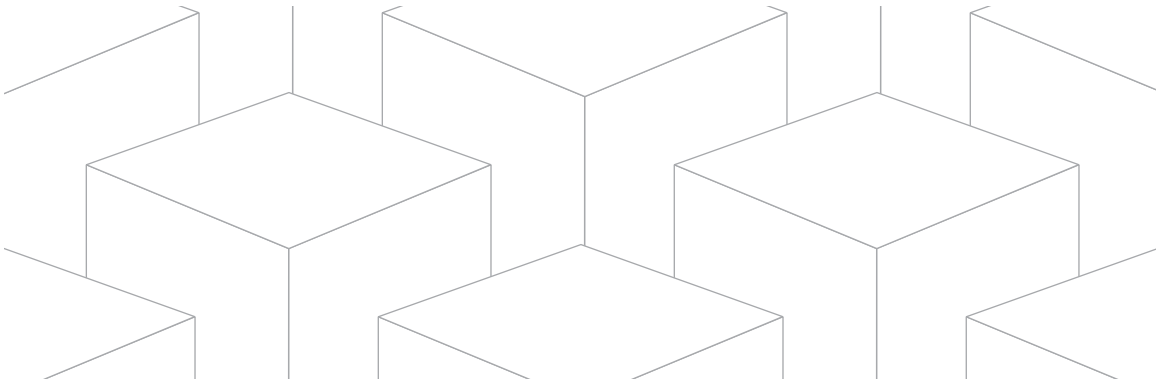


Table of Contents

Introduction

The challenge

For the enterprise IT architect, these are exciting and interesting times indeed. In less than a decade, global commerce has been transformed by Web technologies and the inventive momentum of the information revolution, giving rise to the powerful and dynamic world of e-business.

The popularity and accessibility of the Internet has spurred the development of open, platform-independent technologies and standards for messaging, data encapsulation and application development — such as XML, SOAP, UDDI and, of course, Java. More and more, companies are leveraging these technologies to bring greater efficiency to business transactions and communications — and extending this to partners, customers and suppliers. This, in turn, has paved the way for a burgeoning, planetwide electronic marketplace and a growing reliance on Web-based applications, including the increasing deployment of Web services. The resulting diversity of potent, vital information systems for the enterprise has propelled the role of IT architect to the forefront, much in the same way that the industrial revolution and the complex physical structures it required magnified and reaffirmed the role of the building architect a century and a half ago.

Fawcette Technical Publications, publisher of *Enterprise Architect* magazine, has estimated that there are up to 250,000 enterprise IT architects in the United States. The abundance of professional literature currently published on IT architecture-related issues indicates that the architect's role within the business continues to gain importance.

The architect's role

To mitigate business risk associated with application failure, the IT architect must act during the architectural design process to ensure the availability and health of Web services and applications. Furthermore, both the architect and management must realize that this critical role does not end once the IT architecture has been established and communicated, and the Web application has gone into development. On the contrary, safeguarding the enterprise's ability to optimize Web application performance is a continuous process that extends beyond all phases of development and production.

Therefore, the role of the architect is not merely a technological one. Rather, the central objective of any architecture — and of the Web applications deployed under that architecture — is to achieve the organization's strategic goals across all of its business functions. Consequently, in order to address the architectural issues inherent in Java application management, the architect must call upon a wealth of technical, behavioral and business competencies.

It is important to understand that the architect's role in this context cannot be defined simply as a list of tasks. While it is inevitable that such tasks will be assigned to the architect within the framework of his or her job definition, the function of IT architect is essentially a leadership and governance role and, as such, is objective rather than task-oriented. Thus, while the tasks through which IT architects accomplish their respective functions may vary considerably from one enterprise to another, the business objectives that define their role will tend to be more constant.

Many IT architects will say that the underlying business principles that govern their roles in the management of Java applications and Web services are no different from those that apply to the development of traditional applications in non-Web environments. For this reason, this paper will first examine the role of the IT architect in application management generally and then focus on the specific considerations that are unique to Java-based Web applications.

©2003 Wily Technology, Inc.

What Do IT Architects Do?

Is there a "universal architect"?

Many business professionals — both inside and outside the technological sphere — will discuss the function of the IT architect as if it were a governed by a standard job description. Such comments seem to indicate that architects are thought of as a homogeneous community, with a uniform purpose, skill set and task list.

But how do architects see themselves? The several architects interviewed in the course of preparing this paper spoke of their profession in generic, universal terms, though were clearly aware of the business and technological differentiators of their respective enterprises and positions. (It should be noted that while all of the interviewees described themselves as IT architects, many of their job titles did not actually include the word "architect.") Interestingly, however, when asked to define their role, their responses were as diverse as their individual personalities or the business profiles of their organizations. Here are some of the primary functions that these architects attributed to their position:

- **Technological advocate** — Promoting the cause of technology, and of specific technologies, in order to achieve buy-in from the enterprise's management and IT communities.

  "From my perspective, the architect would have to champion the cause of all things technological. Selling the technological issues to other techies is not really a big deal — it's all about turning around to the business folks and selling them on the technological direction that will best serve the business."

  — Robert, senior technical specialist at a major securities exchange

- **Methodologist** — Selecting, defining and enforcing a consistent, uniform set of products, vendors, methodologies and standards throughout the IT organization.

  "I would say the IT architect defines the cohesive information technology infrastructure across the scope of the enterprise. He or she is responsible for leveraging infrastructure across multiple functional areas, making sure that new technology is appropriately applied — and, of course, with operational concerns such as performance management, availability and reliability."

  — Greg, IT architect with a large, multi-channel retailer of household goods, appliances and furniture

- **Executor of business goals** — Translating the business objectives formulated by management into technology standards that can achieve them.

  "The architect is there to make sure the business goals are being met by the infrastructure. To that end, the architect tries to get involved as early as possible in the design process and make sure the application is going to be both written and released in a way that meets business objectives."

  — Joe, IT architect at a leading global financial services company

- **Facilitator** — Fostering communication among the IT groups, to ensure that IT endeavors reflect a unified, cohesive understanding of the enterprise's business and technological objectives.

  "It's up to the architect to ensure there is training and communication for the developers and get their buy-in and feedback on the architecture. It's not as if architects were the only ones that can understand the crucial issues; developers are intelligent people with great ideas on how to do things even better. Get that feedback and make sure it's all worked into your architecture."

  — Dale, lead member of the technical architecture staff of a major service provider in the transport and logistics industry

©2003 Wily Technology, Inc.

A multidimensional role

Though these focal points may seem to be different, they can be interpreted as different views of the same multidimensional role of the architect within the enterprise, which can be articulated thus:

The architect's role is to assimilate the business goals of the enterprise; to seek and identify technologies and technological solutions that can best achieve them within the total business context; to gain the endorsement of management for their adoption and deployment; to articulate and enforce process, technology and interface standards that support these choices; and to encourage the IT organization to communicate and collaborate internally and externally in the implementation of technological endeavors that will support the business goals.

In the framework of this role, the architect might perform such tasks as:

- Articulating and maintaining architectural objectives and metrics that state precisely which business goals the architecture is designed to achieve, as well as the means by which success will be gauged. Communication of these deliverables may be, for example, in the form of training, documentation, support forums or informal consultation.
- Identifying and justifying potential investments in systems and technologies that can achieve both architectural objectives and business goals, including the solicitation and analysis of information from technology vendors.
- Creating and maintaining a road map for transitioning from current architectural frameworks to the target architectures.
- Establishing, communicating and enforcing application design patterns and development guidelines that are strongly aligned with specific business goals — for example, designing a repository of reusable components in order to save development costs.
- Overseeing and reviewing the application design process to ensure that it meets both the technical constraints of the architecture and the business objectives of the enterprise in terms of reliability, scalability and serviceability.
- Communicating and presenting to the development community the armory of selected tools for application development, management and quality assurance, and providing them with the resources required to master them.
- Resolving clashes and inconsistencies that may arise between the goals of individual projects or between the goals of a specific project and those of the organization, where such conflicts derive directly from architectural constraints.
- Maintaining ongoing involvement in the design and implementation progress of each application, from definition of business requirements through design, testing, deployment and management.
- Overseeing application testing to verify, monitor and ensure performance and scalability, including the identification and alleviation of bottlenecks.

Definition in the negative

The architect's agenda may not include all of the activities cited above — or it may include many more — depending on the nature and business objectives of the enterprise, and on the particular architectures that have been chosen to attain them. Be that as it may, in discussing the duties they perform, architects frequently draw a clear distinction between tasks that they consider to be within their true sphere of responsibility and other tasks — typically support roles — that have devolved upon them by default.

©2003 Wily Technology, Inc.

Plainly stated, many architects find themselves significantly occupied by tasks that they feel are not their job. Some may go as far as to propose that diverting their energies to these tasks may be diluting the effectiveness of their function in achieving business goals. Such activities may include:

- **Development support** — Coding and recoding applications in order to eliminate unforeseen performance issues during development and testing.

  "Very frequently, architects end up doing a lot of things they shouldn't have to — even coding applications. Sometimes that falls into the category of 'It's easier to just do it yourself than to delegate it out.'"

  — Greg, IT architect

- **Deployment support** — Aiding in the development of a blueprint for the transition from testing to production in order to best equip help desk and application support personnel to deal with application performance issues.

  "When the application is complete — when all the performance testing is done, the design is finished, the coding is complete and it's ready to be launched into production — I find that defining the process of moving it from the test space to the production space is hampering my overall responsibility to the organization. Sure, there should be an enterprise-wide blueprint that lays down the foundation for these things, but I shouldn't have to create one at the onset of every application."

  — Robert, senior technical specialist

- **Production support** — Troubleshooting the production environment in order to analyze and diagnose application outages and failures, and resolving conflicts in "finger-pointing" scenarios between IT teams.

  "When there are problems in production, the architect gets dragged in and everyone is saying 'How do we fix this?' The architecture should have been decided many months before production. If something is fundamentally wrong, they look for an architectural solution where you'd hope there shouldn't be one."

  — Joe, IT architect at a leading global financial services company

All of these distractive tasks seem to fall squarely within the scope of application management. Moreover, architects often see themselves burdened by such tasks in the management of high performance, Web-based applications — an area that poses several unique challenges not typically characteristic of traditional application environments. This perception raises the following questions:

- What can the enterprise do to eliminate IT architects' unnecessary involvement in application management functions that are outside their responsibility, thereby maximizing the gain from the architect's function?
- In the context of Java technology-based applications, are there efficient tools or technologies currently available that can ensure the performance of critical components within the architecture and pinpoint potential bottlenecks at an early stage in order to minimize the architect's role in resolving performance crises?
- To what extent should the identification, adoption and deployment of such tools be the responsibility of the architects themselves?

©2003 Wily Technology, Inc.

Application Management Challenges

Achieving efficiencies

As we have seen, architects' competency and familiarity with a broad range of IT systems can cause them to feel that they are diverted from their mission-critical tasks by application management duties that may not be part of their planned responsibility. Because of the architects' crucial role in ensuring the performance of their IT infrastructures — and, by extension, of their entire organization — such disruptions can have a powerful detrimental effect on the success of the enterprise.

The good news is that this unique vulnerability is offset by a unique opportunity — namely, architects' ability to preclude such distractions by identifying, selecting and deploying enterprise application management technologies that ultimately enhance the efficiency of their own performance. In other words, by choosing solutions that can eliminate unforeseen application management crises, the architect is ultimately empowered to focus on accomplishing his or her primary objectives.

To achieve this vision, the architect must examine the three primary pillars of system management, and identify and deploy enabling technologies that optimize the performance of all of them. These three pillars are: products, processes and people.

Optimizing products

In any distributed system, application management can only be effective if application support personnel can continuously monitor the performance of the critical hardware, software, database and networking products that underlie the application environment. Should performance slip, support staff must be able to drill down to the level of granularity necessary to resolve problems quickly. Additionally, they need real-time application performance data to integrate directly with enterprise system management frameworks such as BMC Patrol®, CA Unicenter, HP OpenView or Tivoli TEC®.

The architect must seek and identify enabling technologies that provide these capabilities and prescribe their deployment throughout the enterprise. In order to eliminate surprises and setbacks when applications are transitioned into production, these solutions must be put into action as early in the application cycle as the initial development stages. By the same token — and unlike most current-generation development desk tools, such as profilers — the enabling technologies must also be capable of running in a production environment without degrading performance.

The advent of Web application development — and, in particular, Java programming — extends the monitoring challenge even further. For example, the lack of visibility into enterprise Web applications has resulted in a deficiency in monitoring the important connections between the applications and back-end data systems. Consequently, the extension of enterprise applications to the Web has created the need for a new generation of monitoring products. Specifically, monitoring needs to be extended to the application server, the host of the Web application; to the Java Virtual Machine itself, which serves as the critical connection point; and to key application-connected systems such as databases, IBM® CICS™ Transaction Gateway and the IBM WebSphere® MQ connector™.

The section of this document entitled "Management of Java-Based Web Applications" discusses in greater detail some of the product considerations that are specific to the Java application management environment. It proposes an enterprise Web application management solution that can address them, based on industry-unique Web application monitoring and management tools from Wily Technology.

©2003 Wily Technology, Inc.

Optimizing processes

Central as performance monitoring capabilities are to the application management function — and therefore of prime importance to the architect — visibility alone is not enough to accomplish enterprise application management. The performance data gleaned from monitoring tools must be supported by relevant application management processes, such as operations monitoring, capacity planning, deployment testing, problem determination and exception handling, as well as the establishment of effective alert and remediation procedures.

We have already seen that the articulation and enforcement of process standards is an essential part of the architect's role. No less crucial is the selection of application management tools that can optimize problem analysis processes by supporting them with clarity and proof.

As with the optimization of products, the definition and deployment of application management processes has been strongly affected by the introduction of Java and other Web technologies to the realm of application development.

In short, the enterprise IT architect must select application management tools that are capable of creating highly customized performance dashboards — paralleling specific existing application management processes that are an integral part of the enterprise architecture. Furthermore, such tools must facilitate the extension of application management processes to accommodate the unique requirements of the Java application environment that are presented later in this document.

These requirements point to the need for application management tools that are based on a new approach. The required solutions must be more agile than traditional monitoring tools, which achieve external views of overall application performance from the end user's vantage point. Rather, tools are needed that yield internal views of system performance by taking measurements of key interaction points inside the application modules themselves. This approach, known as the instrumentation method, is employed by Introscope® from Wily Technology and its supporting products, and will be discussed later in greater detail.

Optimizing people

To ensure the availability, performance and control goals that IT and line-of-business management set for each system component, application management is often undertaken as a team activity. This is particularly true of Java applications, because their characteristic intricacy and free-form architecture make performance and availability management too great a task for any single person.

Developing and managing production Java applications involves developers, architects, operators, QA test engineers, database administrators, and application server administrators, various network administrators, outside partners, vendors and suppliers. When a performance or availability issue arises — either during development and testing or after the application is deployed into production — best-in-class IT organizations execute the problem resolution processes that are a vital part of the enterprise architecture, in order to get the application performing again. Such processes typically escalate most issues to a second-tier application support manager (ASM) — someone with an understanding of the specific application as well as of Java, distributed architecture, application server configuration and other key technologies.

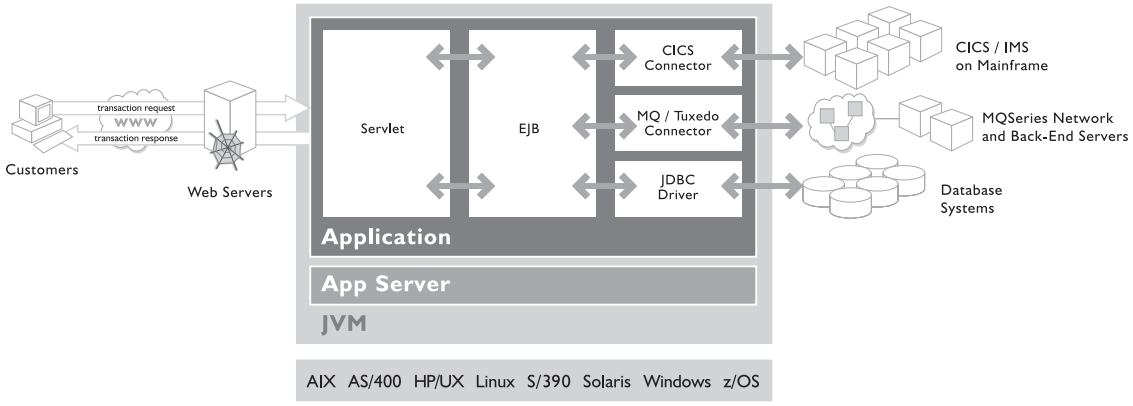
As we have already seen, if adequate resolution processes do not exist or are unsatisfactory, the de facto role of second-tier ASM may devolve upon the architect, diluting her or his ability to perform legitimate architectural functions. Moreover, if problem resolution processes do not include enabling tools and technologies that are shared across the whole team, the situation may quickly deteriorate into incessant infighting, finger-pointing and fruitless experimentation in order to attribute the problem to a particular group of individuals. This has been referred to in the industry as the Blame Game.

©2003 Wily Technology, Inc.

To optimize the human interactions that ensure application control and performance — and to safeguard their own availability to fulfill their designated roles — architects must select and implement common tools that can be used by a broad base of IT teams and bridge the gap between development and operations. However, specialty tools such as profilers, pingers and network monitors, crucial though they may be to system management, cannot fulfill these requirements when used alone.

What is needed are tools that can furnish a shared view into the whole application, in order to foster a consistent vision and a common language with respect to the hard information that attests application performance. Such tools must be capable of conveying detailed performance metrics on critical components while remaining lightweight enough to run in production without degrading performance. The integrated family of enterprise Java application management products from Wily Technology fulfills this need.

The Whole Application™ View



©2003 Wily Technology, Inc.

Management of Java-Based Web Applications

The black box

The mass migration of large-scale enterprises from mainframe and client/server systems to Web technology has extended the user base of enterprise applications from employees-only to business partners, suppliers and to the customers themselves. But with this new architecture comes a new challenge: the total application management and control that had been established for traditional enterprise architectures has not kept pace with the shift to Web applications. Simply put, the extension of enterprise applications to the middle tier of the Web application server and to browser-based clients has created a blind spot of systems management visibility — in essence, a black box.

The lack of total application visibility has made it harder than ever for operations personnel to ensure quality of service to their constituents. Specifically, Web applications can fail at the critical moment of deployment, due to problems that were not detected during development. There are several possible reasons for such failures:

- Deployed applications might be exercised in ways that could not have been anticipated in the development cycle.
- Real load conditions are almost always unknown, meaning they cannot truly be simulated in development.
- Interactions with production systems may be different from development simulations.
- There may be an unforeseen communications problem between network monitoring software, enterprise management software and application-level monitoring software.

To mitigate the risk that SLAs could be breached by performance failures, the enterprise architect must identify, adopt and deploy enabling technologies that can deliver visibility across the whole application environment while combining application performance data with other network management data from existing enterprise systems management sources.

Application management and the JVM

The traditional armory of application management solutions — including individual monitors for application servers, database resources and transaction systems — falls short in the Web application environment. This is because the major hub of activity within the Web application — the Java Virtual Machine (JVM) — is invisible to such systems.

Whether physically located on a separate machine or embedded as another logical machine within the mainframe, the JVM is the central switching yard for enterprise data movement and for customer-facing enterprise applications. Typically, the JVM moves data across the following system components:

- Transaction management systems
- Message queuing systems
- Enterprise database resources
- Web application servers.

Because traditional management tools do not offer visibility into the production JVM, the application and the application servers on which they run, they cannot deliver effective systems management in the Java environment. Architects must supplement these tools with a management solution that offers production monitoring across the entire application, at the component level, while integrating with existing enterprise system management solutions and processes.

©2003 Wily Technology, Inc.

Blame revisited

As previously discussed, problem resolution procedures that are not properly managed or that are not supported by the appropriate tools can result in endless Blame Game scenarios. But in fact, the notion of blame can actually be a positive one when it refers not to indiscriminate finger-pointing, but to a systematic, informed effort to legitimately isolate the precise component causing the problem.

The ability to place blame electronically, using a smart management platform that automatically and accurately identifies performance bottlenecks, can help resolve Web application problems more quickly and is therefore a compelling goal for the IT architect. Deployment of such a platform as part of the enterprise architecture can streamline the use of maintenance resources while eliminating the finger-pointing among IT staff.

Conversely, the ability to rule out a particular element of the IT infrastructure as the cause of a performance or availability issue can be a valuable advantage in negotiating a fix with external partners or with other departments within the organization. For example, being able to show that there are spikes in external response time, yet no spikes in the application's internal response time, can help focus attention on an outside data source.

Instrumentation

A next-generation approach to these issues involves instrumenting Web applications by automatically installing a set of lightweight probes within the application that log and monitor the real-time transactions that it executes. Because these probes require no changes to application source code, it is possible to set up, customize and install them as instrumentation elements within the monitoring tool, residing at the interfaces to the various dependent modules, networking nodes and databases that contribute to the application.

An application management solution based on instrumentation can identify performance bottlenecks in real-time down to the fine-grained component level, alert the correct specialist in keeping with established resolution procedures, initiate an automated task to rectify the problem and verify that other components within the application are working properly. (See Figure 1.)

Most enterprise network and systems management solutions excel at determining whether network elements are alive or dead as well as whether they are exceeding rudimentary thresholds, but they cannot deliver the degree of granularity required in order to effectively monitor and manage Java-based Web application performance at the individual component level. Other offerings may be ideal instrumentation products for application development, but are ineffective in the deployed application space because of the performance degradation they tend to cause.

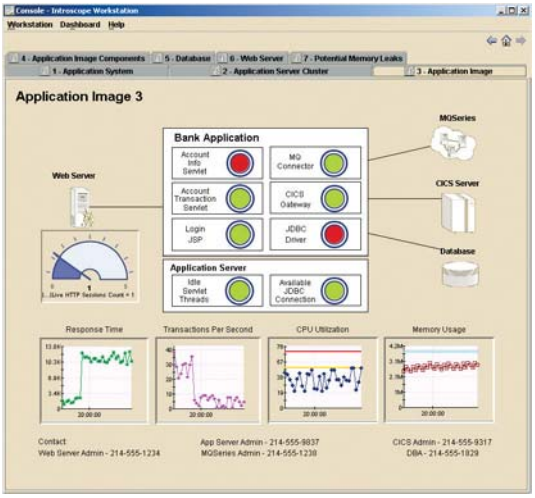


Figure 1. Introscope from Wily Technology, features configurable dashboards than can monitor a production application while showing detailed performance metrics about critical components.

©2003 Wily Technology, Inc.

Wily Technology's application management solution

For Java-based Web applications, Wily's Introscope and supporting products bridge this gap between the requirements of development and production environments. Introscope effectively employs instrumentation technology in a near-zero overhead application management solution that automatically monitors the performance of such Java components as servlets, EJBs, JSPs, and JDBC data sources. By including Introscope within the enterprise architecture, IT architects can provide operations personnel with an effective tool to help ensure the availability of production Java applications, to fine-tune them for better performance and to accurately identify both internal and external performance failures.

In addition to providing this visibility, Introscope features extensions-called "PowerPacks,™" which monitor the interaction between the JVM and the transaction processing system (such as IBM CICS), the messaging middleware system (such as IBM WebSphere MQ) and the database. These extensions bring the critical element of asynchronous time measurement down to the granularity of the SQL call, the WebSphere MQ message and the individual CICS transaction, bringing clarity and proof to problem analysis that would otherwise be little more than educated guesswork. Introscope can present its data to a Management Information Base (MIB), enabling other enterprise management applications and consoles to receive live application performance data and alerts while utilizing the MIB for more detailed information and cross-correlations.

With customizable dashboards, Introscope can be tailored to the enterprise's unique needs to trigger specific actions, analyses and procedures. The application program interface (API) that enables the MIB also allows Introscope to receive information from other sources, such as JMX-compatible Web application servers, as well as from the PMI interface of IBM WebSphere Application Server. These capabilities enable the architect to provide the foundation for a comprehensive Web application management process for the enterprise.

Conclusion

IT architects perform a key role in achieving the business goals of the enterprise. This role extends beyond the articulation of technological solutions and standards, embracing organizational methodology as well as interpersonal collaboration and communication. There is much flexibility in the repertoire of tasks that architects may undertake in order to achieve their business objectives and those of their organizations. But regardless of what these specific assignments may be, it is also the architects' job to minimize disruptions to their own effectiveness, in order to maximize the contribution they can make to their organization. Thus, the architect is charged with two related objectives — furthering the business goals of the enterprise through technological solutions and ensuring the focus of his or her own contribution to the organization.

The potential for success in achieving these objectives is particularly high in the realm of application management, specifically in the context of Java-based Web applications. To realize this potential, the architect must identify and deploy enabling technologies that can ensure that each component within the architecture — products, people and processes — performs as it should.

©2003 Wily Technology, Inc.

About Wily Technology

Wily Technology, Inc. has become the leader in performance management software for Enterprise Java Applications by helping our customers keep their applications running at peak performance.

Wily enables Global 2000 organizations to pinpoint performance problems in their mission critical software systems before they impact customers. Wily's products give IT operations and support teams the ability to see inside production Java applications and the whole application environment, a vital step for improving the performance and reliability of Web operations.

Wily's comprehensive solution called Wily 4 gives customers visibility into their new generation Java applications, integrates into existing IT systems management processes, and when combined with Wily professional services and best-practices, helps enterprise IT personnel become world-class managers of Web applications.

Wily has established strategic partnerships with industry leaders such as BEA Systems, Hewlett-Packard, IBM Corporation, Oracle and Sun Microsystems. The result of these close development relationships is that our customers are assured of the ability to monitor and manage applications built on the latest application server technology.

For more information about Wily Technology, Introscope, or any of Wily's family of enterprise Java application management products, visit www.wilytech.com.

Wily Offices Around the World

Wily Technology, Inc.
8000 Marina Boulevard, Suite 700
Brisbane, CA 94005
U.S.A.

1 800 GET WILY <U.S. toll free>
+1 415 562 2000 <direct>
+1 415 562 2100 <fax>
info@wilytech.com

Wily Technology (UK) Ltd.
Knyvett House
The Causeway
Staines, Middlesex
TW18 3BA
UK

Phone: +44 1784 895 158
Fax: +44 1784 895 075
sales-emea@wilytech.com

Wily Technology Asia / Pacific
Phone: +61 2 6286 9663
Fax: +61 2 6286 9664
sales-asiapac@wilytech.com