



Wily Introscope® for Microsoft .NET

*Managing Complex Web Applications in a
Production Environment*

White Paper

Abstract

Enterprises are increasingly relying on Web-based applications to perform mission-critical functions in service-oriented architectures. Consequently, managing these applications and quickly resolving their performance issues are key to business success. Microsoft® .NET is a popular strategy for implementing Web applications and Web services because it offers an extremely flexible development and execution environment—the Microsoft .NET Framework. Wily Introscope® for Microsoft .NET from CA helps manage the performance of applications that are based on .NET and on other technologies. It continuously monitors complex Web applications in production environments, in real time, with low overhead. It provides transaction visibility and diagnostic capabilities across the entire Web infrastructure to resolve problems before they affect customers. This paper explains to IT administrators and other operations professionals how Introscope simplifies the challenge of managing complex Web applications, ensuring superior delivery of IT services to end users.

Contents

Introduction	1
Microsoft .NET Supports Mission-Critical Applications	1
Advantages of Microsoft .NET	1
Wily Introscope Helps Manage .NET Applications.....	1
The Challenge of Managing Web-Based Applications	3
Essential Elements of Microsoft .NET	4
Multi-Language Support from a Common Language Runtime	4
Rich Web Pages from ASP.NET	4
Comprehensive Data Access from ADO.NET.....	5
Support for Scalable, Flexible Three-Tier Architectures	5
Why the Three-Tier Architecture Is Important	6
Why Managing Three-Tier Applications Can Be Challenging.....	6
How Wily Introscope Solves the Complexity Challenge	7
Continuous Application Monitoring with Low Overhead.....	7
Deep Visibility into Applications and Transactions.....	7
Customizable Dashboards for Technical and Nontechnical Stakeholders	8
Rapid Identification and Triaging of Performance Problems.....	8
Real-Time and Historic Performance Metrics	9
Support for Heterogeneous Environments.....	9
Support for Existing Management Procedures	9
Conclusion	10
About CA and Wily.....	11

Introduction

Mission-critical Web applications in today's service-oriented architectures (SOAs) perform many important business functions: revenue generation, supply chain management, and the delivery of essential services, to name just a few. For organizations deploying these Web applications, the success or failure of their business depends on:

- End users and customers completing transactions reliably.
- IT teams finding and fixing IT problems rapidly, before service level agreements (SLAs) are breached.

Organizations are increasingly making Microsoft .NET connection software an integral part of their IT environments because .NET technologies support rapid development of enterprise-ready Web applications. As a result, IT administrators and other operations professionals need tools to continuously manage and monitor their .NET applications in production environments and to make sure those applications deliver optimum performance and service to users and customers.

Microsoft .NET Supports Mission-Critical Applications

Microsoft .NET is the Microsoft strategy for connecting systems, information, and devices through simplified application development and Web services. The Microsoft .NET Framework provides a foundation for implementing the .NET strategy. As a flexible development and execution environment, the Framework allows different programming languages and libraries to work together to create mission-critical applications that are easier to build, manage, deploy, and integrate with other networked systems.

Advantages of Microsoft .NET

.NET is an ideal strategy for SOAs—an IT model that reflects the dynamic nature of business by using Web services to support flexible, loosely coupled systems and cross-platform solutions. .NET technologies can be used to integrate very disparate computing environments. It frees organizations from the confines of proprietary technology, providing the flexibility and scalability to connect existing IT systems and build a foundation for highly adaptable computer technology.

.NET can reduce operating costs by helping to connect previously unconnected systems. It helps employees access information when and where they need it, integrates services and applications with customers and partners, and lowers the overall cost of IT by supporting tools that help developers quickly create new solutions.

Wily Introscope Helps Manage .NET Applications

Wily Introscope® for Microsoft .NET helps IT administrators and other operations professionals handle the challenges of managing modern, diverse, heterogeneous IT infrastructures. It delivers real-time monitoring of mission-critical .NET applications, including applications that are built on the .NET Framework and applications that connect using the protocols of Web services, in a production environment.

To help IT administrators meet their application performance management (APM) responsibilities, Introscope offers essential features that many traditional management solutions don't:

- Introscope monitors performance at the application and transaction level, not just the overall performance of discrete systems such as databases and Web servers.
- Introscope continuously manages the entire Web infrastructure, in real time and with low overhead.

- Introscope can be deployed immediately and without development effort because no recoding of existing applications is required. It has the ability to monitor custom .NET applications that were never designed with APM in mind.
- Introscope installs unobtrusively into the IT environment, runs easily alongside other management tools, and works seamlessly with the .NET Framework itself.
- Introscope helps IT administrators communicate critical performance data throughout the enterprise. It provides a common language that everyone from senior management to developers can use to ensure that applications are achieving performance and business goals.

Introscope for Microsoft .NET helps simplify and support managing environments that deploy applications based on .NET and on other technologies, too. It also helps IT staff detect and resolve application performance problems before they affect users and customers.

The Challenge of Managing Web-Based Applications

As today's organizations increasingly move essential business functions to the Web, the requirements for the usability, reliability, and efficiency of Web-based applications have become very strict. Failures or slowdowns, even for brief periods, can be costly. Optimizing the performance of applications requires close management and monitoring by IT departments.

However, the highly interconnected application environments of Web-based systems and SOAs pose special challenges for performance management. Web applications tend to be built from discrete components that are distributed across a variety of systems. The resulting Web applications offer enormous flexibility, but they also increase the complexity of the IT environment in ways that traditional APM approaches often cannot handle. Also, older custom Web applications were often designed without taking advantage of features that make managing and monitoring of performance easier.

Without the right tools at their disposal, IT administrators cannot monitor application health in production environments, cannot proactively detect performance issues, and cannot gather essential root-cause data. If latency or errors occur as transactions flow through multiple systems, locating the source of the problem may be difficult or impossible. One reason for this is that traditional APM approaches view the IT infrastructure only from a fairly high-level functional perspective, looking at the overall performance of each "silo" (discrete components of the IT infrastructure such as databases and Web servers).

Although monitoring the performance of individual silos can be very helpful, Web application transactions cross a wide number of these technology silos, and each one has an impact on the overall application performance. Administrators need an APM solution that can look deep into each technology component, while also following the entire transaction path, and Introscope for Microsoft .NET has this capability. But to understand how Introscope handles these challenges, it is helpful to know about the architecture and some of the essential elements of Microsoft .NET.

Essential Elements of Microsoft .NET

Microsoft .NET supports quickly building, deploying, managing, and using connected security-enhanced solutions. This is among the many reasons that enterprises are adopting it for their mission-critical applications. The Microsoft .NET Framework is a programming model and runtime for implementing .NET solutions, and it is an integral component of the Windows® operating system for creating Web applications and Web services.

The .NET Framework includes prewritten code provided in class libraries (ASP.NET, ADO.NET, Windows Forms, and XML classes) that developers can combine with their own code to build powerful applications. Using prewritten code saves development effort and helps ensure that the applications will be more secure, error resistant, and well behaved. In addition to class libraries, the Framework includes a Common Language Runtime (CLR) that provides error handling, memory management, and other services.

Multi-Language Support from a Common Language Runtime

The CLR is a managed execution environment that handles an application's runtime requirements so that developers won't have to worry about them. The CLR makes it easy for developers to design components and applications whose objects interact across programming languages. For developers, this means they can choose the language or languages that they know best or that will most efficiently implement their applications' goals. For IT administrators, this means that applications written in a variety of languages by different teams across the enterprise will work together and can be managed and monitored seamlessly.

How does it work? The CLR includes enhanced security, versioning, and deployment control; a simplified model for component interaction; and debugging and profiling services that allow objects written in different languages to communicate with each other. This cross-language integration is possible because language compilers and other tools that support the CLR use a common type system defined by the CLR, and they follow consistent CLR rules. Different compilers expose CLR functionality in ways that are intended to be helpful and intuitive to developers; how developers experience the CLR is optimized for the specific compiler or tools they choose. Microsoft Visual Studio®-branded products are particularly well-suited for Framework development because they support the full range of CLR features.

Rich Web Pages from ASP.NET

ASP.NET includes the services necessary to build enterprise-class Web applications, dynamic Web sites, and XML services with a minimum of coding. It is a class library and unified Web development model within the .NET Framework that includes a page-and-controls framework that supports encapsulating common user interface functionality in easy-to-use, reusable controls. These controls are similar in concept to the controls in Windows—making ASP.NET an ideal resource for developers who are transitioning to the .NET Framework from traditional Windows programming—but the controls are much more flexible.

ASP.NET provides features to control the appearance of a Web site through master pages, themes, and skins that developers can define and apply at a site level, a page level, or a control level. For IT administrators, this means that an organization's Web applications can provide a consistent look and feel, which improves the applications' usability.

ASP.NET Web pages can be requested from any browser or client device. Developers can use the same page for multiple browsers or design a page to target a particular browser, such as Microsoft Internet Explorer®, to take advantage of specific features in that browser. ASP.NET also supports mobile controls for Web-enabled devices such as cell phones and handheld computers. For IT administrators, that means the organization's mobile applications can be built using the same

technology as any other Web applications and thus managed in the same way—including using Introscope.

ASP.NET includes all the components that developers need to create Web applications that satisfy an organization's usability, performance, and APM goals:

- A page-and-controls framework
- The ASP.NET compiler
- Security infrastructure
- State-management facilities
- Application configuration
- Health monitoring and performance features
- Debugging support
- An XML Web services framework
- Extensible hosting environment and application life cycle management
- An extensible designer environment

Like all the class libraries in the .NET Framework, ASP.NET provides multi-language support and many prewritten controls and tools and that reduce development effort. For example, a developer can easily create a rich Web page that displays database records because ASP.NET contains a variety of data access controls and is tightly integrated with ADO.NET.

Comprehensive Data Access from ADO.NET

Nearly every mission-critical application in an enterprise needs to access one or more databases, and ADO.NET provides a rich set of components for creating distributed, data-sharing, and data-consuming applications. This class library can access relational data, XML, and application data. ADO.NET supports several development needs, including the creation of front-end database clients and middle-tier business objects used by applications, tools, languages, or Internet browsers.

ADO.NET was designed hand-in-hand with the XML classes in the .NET Framework—both are components of a single architecture. Through ADO.NET, applications have reliable access to data sources such as Microsoft SQL Server™ as well as other data sources exposed through standard protocols. A data-sharing consumer application can use ADO.NET to connect to these data sources and retrieve, manipulate, and update data within them.

In the past, data processing applications have relied primarily on a connection-based, two-tier architecture. However, developers are increasingly using three-tier architectures to provide better scalability for their applications, and the .NET Framework—with its robust data access functions like those in ADO.NET—readily supports this transition.

Support for Scalable, Flexible Three-Tier Architectures

Enterprise-class Web applications should provide the detail and rich experience of a desktop application, even though they run through a simple Web browser. As a step toward reaching this goal, enterprises often design their Web applications using a three-tier architecture.

In a three-tier client/server architecture (see Figure 1), the application's user-interface layer runs on a client computer, and the application's code layer, which defines what the application does and how it works, runs on an application server. Services such as communications, file access, printing, and database run on other servers; the most important of these from the development perspective is database services. Messages between the tiers are exchanged using standard protocols.

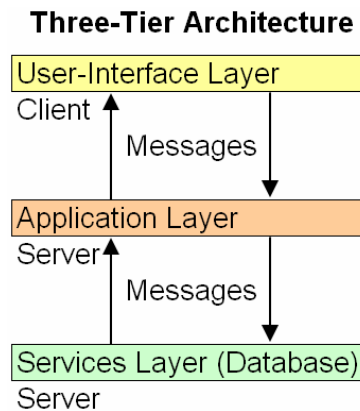


Figure 1. Separating an application's architecture into tiers provides scalability and other benefits but increases complexity.

The three-tier approach is more complicated than running all three layers on the client computer (one-tier) or running the user interface and application layers together on the client and running the services layer on a separate server (two-tier). But the important advantages that the three-tier architecture offers compensate for the increased complexity.

Why the Three-Tier Architecture Is Important

The modern three-tier application architecture provides many benefits over simpler architectures, including reusability, flexibility, maintainability, and scalability. With the three-tier approach, developers can share and reuse the components and services they create and can distribute them across a network of computers as needed. This makes it easier to divide large and complex projects into simpler ones that can be assigned to different programmers or programming teams. Organizations can also deploy components and services on a server to help keep up with changes and can redeploy them as growth of the application's user base, data, and transaction volume increases.

Traditionally, three-tier applications have been more difficult to design than one-tier or two-tier applications, requiring more development effort and more actual coding. However, the .NET Framework greatly simplifies this process because of its extensive class libraries, especially for data access tasks, and because of its interoperability with so many languages and systems. Also, the software that is installed on users' computers to support the .NET Framework works closely with server-based .NET applications to provide a rich client experience.

Why Managing Three-Tier Applications Can Be Challenging

SOA-based enterprise applications don't run alone. They request actions from other applications and services that, in turn, request other applications and services, and so on. Transactions and messages flow between layers within an application and also between applications. In real-world production environments, the application message flow is far more complex than the four message arrows shown in Figure 2.

The three-tier application architecture is sometimes called *n-tier* to indicate that these application environments can grow quite large (although the same three logical tiers still define each application). And as the number of applications grows linearly, the complexity of their interactions can grow exponentially. This makes the process of tracking down the sources of performance problems very challenging. However, addressing this challenge is precisely what Introscope for Microsoft .NET does.

How Wily Introscope Solves the Complexity Challenge

IT administrators are responsible for making sure their customers get consistent, reliable access to their organization's IT resources. When something goes wrong, they have to restore service quickly. Introscope manages .NET and other applications 24 hours a day, seven days a week, with low overhead, across heterogeneous technologies and operating systems. When problems occur, it identifies which ones affect users and customers most, helping administrators to triage the problems based on severity and to identify the right owners to solve them.

Continuous Application Monitoring with Low Overhead

Once deployed, Wily Introscope for Microsoft .NET becomes an integral part of the IT environment, providing real-time intelligence about application performance. This true and uninterrupted monitoring of the production environment is a critical capability for ensuring superior delivery of IT service.

Introscope automatically adds probes to .NET applications through a standard Microsoft interface. Consequently, Introscope can be deployed rapidly without developers having to write special code for APM purposes. And the use of a standard interface results in low system overhead, so the continuous monitoring itself causes almost no reduction in the Web applications' performance.

Deep Visibility into Applications and Transactions

Introscope provides comprehensive views into each .NET application and identifies performance problems whether they're inside the CLR environment or in supporting systems such as databases (see Figure 2).

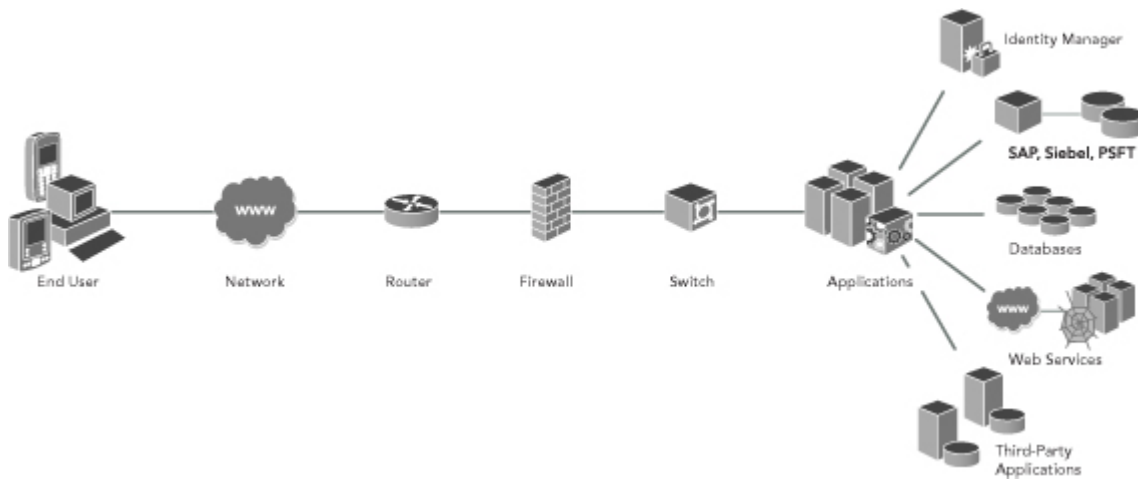


Figure 2. The deep visibility of Introscope into the .NET Web application environment helps to identify bottlenecks, rapidly resolve problems, and meet performance and business goals.

Introscope has the ability to automatically discover the components that .NET applications rely on, such as those in ASP.NET, Active Server Methods (ASMX), ADO.NET, Enterprise Services, Directory Services, and .NET Messaging. For example, Introscope SQL Agent provides visibility into ADO.NET all the way down to the individual SQL statements. By identifying the relationships among these .NET components, Introscope can follow the flow of transactions through multiple systems. It can isolate specific slowdowns and provide detailed information for resolving them.

Introscope also monitors the Windows environment through integrated Perfmon metrics, helping IT administrators understand the relationship between the Windows environment and the performance

and availability of the .NET application. It monitors Web services produced and consumed by .NET applications, including live views of individual transactions in which Web services are involved, the number and nature of Web service faults, and component interactions.

Moreover, Introscope can be easily customized to monitor third-party application code and to display the results in customizable dashboards.

Customizable Dashboards for Technical and Nontechnical Stakeholders

Easy-to-use dashboards in Introscope present intuitive views of the application infrastructure through customizable graphical displays (see Figure 3). However, these dashboards don't merely list raw metrics. With powerful graphing and reporting features, users can create individual dashboards that match the different ways they think about applications and about the IT environment.

Introscope dashboards can integrate seamlessly with predefined operations procedures, making them ideal for operations personnel who need to identify and triage performance issues but who may have little or no expertise in programming or the .NET Framework. In this way, the flexible dashboard features in Introscope support an important organizational goal: to involve all stakeholders—technical and nontechnical—in the APM process.



Figure 3. Wily Introscope for Microsoft .NET provides customizable views of the managed environment.

Rapid Identification and Triage of Performance Problems

When performance problems occur in mission-critical applications, operations and application support personnel need to know at the soonest possible moment. Introscope includes features to gather timely, accurate information about the source of performance problems and to help IT administrators determine where to assign the fix.

- Alerts notify administrators onscreen, through e-mail, or by pager when predefined events occur. Introscope can also use native APIs or .NET messages to handle the events so that notification can be integrated with existing management solutions such as Microsoft Operations Manager 2005.
- Introscope tracks and isolates individual problematic transactions in the live production system across multiple processes. By examining the context of transaction paths, application support personnel can then determine the type of problem and take the next steps for remediation.
- Blame™ technology helps administrators isolate bottlenecks throughout even complex Web applications, all the way down to and including individual components. This enables rapid root-

cause diagnosis so that problems can be fixed and operations restored with minimum disruption to users and business processes.

Rapidly identifying problems and then effectively triaging them helps eliminate the chaos and wasted time of the blame game and finger pointing. Additionally, the intelligent analysis engine in Introscope models application behavior and supports proactive management by predicting some errors before they become performance problems.

Real-Time and Historic Performance Metrics

Introscope includes Wily SmartStor™, which records 100 percent of the data reported to Introscope and makes it available for viewing in customized dashboards or in easy-to-read reports for problem solving, problem replay, trend analysis, and capacity planning. In addition to reported data, Introscope also stores all monitored transactions in a Transaction Events Database™.

The performance of every application component can be viewed in real time, or it can be replayed from any time period for up to a year. This “flight data recorder” feature enables IT administrators to re-create and examine the conditions that cause errors, including intermittent errors that can be especially difficult to observe in real time.

Historical data reporting also allows IT, other departments, and lines of business to manage SLAs with hard data, plan for future needs, and comply with corporate governance initiatives.

Support for Heterogeneous Environments

Introscope is designed for heterogeneous environments. Its continuous-monitoring approach applies not only to applications based on .NET technology running on Windows but also to Java applications running on the Java Platform, Enterprise Edition (Java EE). Introscope for Microsoft .NET is built on the same technology foundation as Wily Introscope for Java EE, so IT administrators can completely integrate the monitoring of the two environments and correlate performance between .NET and Java applications.

Introscope for Microsoft.NET supports a variety of operating systems, including all editions of the Windows Server® 2003 operating system, and it works in both distributed and mainframe architectures. It doesn't require systems management frameworks but can integrate with them using dynamic management information base (MIB) support. It supports existing enterprise security with fine-grained control over user roles and access, integration with architectures based on the lightweight directory access protocol (LDAP), and strong encryption of user authentication.

Support for Existing Management Procedures

Introscope for Microsoft .NET complements an organization's established procedures for application management. It empowers IT departments to create an effective process for detecting and resolving Web application performance issues without having to abandon any tools or procedures that have previously served the organization well.

Beyond IT Operations, Introscope also provides a common language that senior management, middle management, operations personnel, system administrators, quality assurance teams, and developers can use to ensure that applications are successfully achieving business and performance goals. The flexibility and integration that Introscope offers lets every stakeholder use performance data collaboratively for monitoring application health and availability.

Conclusion

The Microsoft .NET Framework provides an ideal environment for rapidly developing and deploying Web applications and Web services, and more and more businesses are deploying .NET applications to perform mission-critical business functions.

The rise of SOAs that use applications built on both Java EE and .NET technologies requires IT administrators to implement proven processes for application performance management. By simplifying the management of complex Web applications, Wily Introscope helps organizations improve the customer experience, ensure superior service delivery, and better achieve business goals.

By adopting Introscope for Microsoft .NET, enterprises can confidently deploy .NET applications, knowing that critical IT management needs and performance objectives will be met.

About CA and Wily



Founded in 1976 and headquartered in Islandia, New York, Computer Associates (CA) is a Microsoft Gold Certified Partner and one of the world's largest providers of IT management software. With 150 offices in 45 countries, the company's more than 5,300

developers create and deliver solutions that unify and simplify the management of IT, securely, across the entire enterprise. CA calls this approach Enterprise IT Management (EITM). The company serves 98 percent of Fortune 1000 companies, as well as government agencies, educational institutions, and thousands of companies in diverse industries.

Wily Technology is a division of the Business Services Optimization unit at CA. Wily produces enterprise application management solutions that enable companies to successfully manage the health and availability of their applications and infrastructure. This includes detecting and diagnosing application slowdowns and failures as well as better assessing the impact of application performance on business success. The results of deploying these solutions include better customer service, more stable revenue streams, and higher IT productivity.

Introscope is the flagship product from CA's Wily Technology Division and the market-leading solution for managing the performance of business-critical Web applications. It provides a highly scalable architecture to help manage complex deployments, intelligent automation to enhance usability, and comprehensive visibility into transactions and business processes for rapid problem resolution. The latest version, Introscope for Microsoft .NET, builds on the success of this innovative software by helping to ensure the performance and availability of applications built on the Microsoft .NET Framework.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT AND CA MAKE NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering this document or the subject matter included in this document. The furnishing of this document does not give the reader any license to these patents, trademarks, copyrights, or other intellectual property.

© 2006 Microsoft Corporation. All rights reserved.

Copyright © 2006 CA. All rights reserved. All trademarks, trade names, service marks and logos referenced herein belong to their respective companies.

Microsoft, Internet Explorer, Visual Studio, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Wily Technology, Introscope, Introscope PowerPack, and SmartStor are registered trademarks or trademarks of CA Inc. All other trademarks are property of their respective owners.

Published October 2006

