

Compliments of **riverbed**

Riverbed Special Edition

Application Performance Management

FOR
DUMMIES[®]
A Wiley Brand

Learn:

- Performance monitoring from the user's view
- How to trace application transactions at the code level
- To apply big data techniques and analytics



Lawrence C. Miller, CISSP

These materials are the copyright of John Wiley & Sons, Inc. and any dissemination, distribution, or unauthorized use is strictly prohibited.

Riverbed Performance Management

Riverbed Performance Management gives IT visibility and actionable insight to deliver the application performance users and the business demand.

RPM's unique combination of end-user experience monitoring, transaction tracing, deep component monitoring, and IT infrastructure/network management maximizes the performance, availability, and productivity of critical applications for:

Complete visibility. Bring together the complete application picture: end user, application transactions, IT, and network infrastructure.

Real answers. Apply "big data" analytics to automate analysis and troubleshooting, dramatically accelerating problem diagnosis and improving IT efficiency.

Right information. Present the relevant information about application performance to a wide range of audiences, businesses, and IT.

For more information about what Riverbed Performance Management can do for you, visit www.riverbed.com/rpm/.

Application Performance Management

FOR
DUMMIES[®]
A Wiley Brand

Riverbed Special Edition

by Lawrence C. Miller, CISSP

FOR
DUMMIES[®]
A Wiley Brand

Application Performance Management For Dummies® Riverbed Special Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

Copyright © 2013 by John Wiley & Sons, Inc., Hoboken, New Jersey

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, the Wiley logo, For Dummies, the Dummies Man logo, A Reference for the Rest of Us!, The Dummies Way, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. Riverbed, OPNET, Steelhead, Cascade, Wireshark, FlyScript, Stingray, and the Riverbed logo are trademarks or registered trademarks of Riverbed Technology Inc. in the United States and other countries. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc., is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Business Development Department in the U.S. at 317-572-3205. For details on how to create a custom For Dummies book for your business or organization, contact info@dummies.biz. For information about licensing the For Dummies brand for products or services, contact BrandedRights&Licenses@Wiley.com.

ISBN 978-1-118-71965-7 (pbk); ISBN 978-1-118-72175-9 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Publisher's Acknowledgments

Senior Project Editor: Zoë Wykes

Editorial Manager: Rev Mengle

Senior Business Development Representative: Sue Blessing

Senior Project Coordinator: Kristie Rees

Special Help from Riverbed: Sue Cole, Heidi Gabrielson, Russ Elsner, Dimitri Vlachos, Rebecca Salie

These materials are the copyright of John Wiley & Sons, Inc. and any dissemination, distribution, or unauthorized use is strictly prohibited.

Table of Contents

Introduction	1
About This Book	1
Icons Used in This Book.....	2
Chapter 1: Understanding What Application Performance Management Is.....	3
Complex Application and Infrastructure Landscape ...	3
Different Elements of APM.....	8
Monitoring and Troubleshooting Applications	9
Chapter 2: From the Driver's Seat: End-User Experience Monitoring.....	13
Understanding User Behaviors and Patterns.....	14
End-User Experience Monitoring.....	16
Real end-user experience monitoring.....	16
Synthetic end-user monitoring.....	17
Chapter 3: Looking Under the Hood: Transaction Tracing.....	19
Following a Single Application Transaction Across All Components.....	20
Monitoring Performance at the Code Level at Every Step.....	20
Tracing Transactions	21
Chapter 4: Horsepower and Fuel Efficiency.....	23
Instrumentation: Keeping It Light	23
Dealing with Lots of Data — Every Transaction	24

Applying Big Data Techniques	25
Leveraging Analytics to Detect Patterns and Problems	27

Chapter 5: Exploring the Application Performance Management Terrain..... 29

Monitor Application Performance and SLA Compliance	30
Perform Triage	31
Troubleshoot Application Performance	31
Map Application Dependencies	32
Manage Application Performance in the Cloud	33
Manage Performance with Application and Infrastructure Change	35

Chapter 6: Ten (Okay, Five) Application Performance Management Evaluation Criteria..... 37

End-User Experience (EUE) Monitoring.....	37
Transaction Tracing and Application Component Monitoring	39
Application Dependency Mapping	40
Analytics	41
Dashboards.....	41

Introduction



Application Performance Management For Dummies, Riverbed Special Edition, introduces you to application performance management (APM) solutions and how these tools can help you monitor and troubleshoot your mission-critical applications — from the perspective of your users, as well as your systems.

About This Book

This book contains volumes of information that rival a map of the human genome, conveniently distilled into six short chapters chock-full of just the information you need! Each chapter is individually wrapped (but not packaged for individual sale) and written to stand on its own, so feel free to start reading anywhere and hop, skip, or jump between chapters (or around your office)! Here's a brief look at what awaits you.

Chapter 1: Understanding What Application Performance Management Is. I begin by exploring some application trends and different elements of APM, including monitoring and troubleshooting.

Chapter 2: From the Driver's Seat: End-User Experience Monitoring. This chapter takes a look at application performance from the user's perspective.

Chapter 3: Looking Under the Hood: Transaction Tracing. Here, you take an in-depth look at how application transaction tracing accelerates troubleshooting, and how APM helps bridge the workflow between application support and development.

Chapter 4: Horsepower and Fuel Efficiency. In this chapter, you learn to use a “big data approach” to deal with all of the data that APM collects.

Chapter 5: Exploring the Application Performance Management Terrain. Here, you learn about some creative uses for APM to help improve efficiency and effectiveness in your organization.

Chapter 6: Ten (Okay, Five) Application Performance Management Evaluation Criteria. Here, in classic *For Dummies* style, I tell you about several important criteria to look for in an APM solution.

Icons Used in This Book

Throughout this book, you sometimes see icons that point out important information. Here's what to expect.



This icon points out information that may well be worth committing to your nonvolatile memory — along with anniversaries and birthdays!



If you're an insufferable insomniac or vying to be the life of a World of Warcraft party, take note. This icon explains the jargon beneath the jargon.



Thank you for reading, hope you enjoy the book, please take care of your writers! Seriously, this icon points out helpful nuggets of information.

Chapter 1

Understanding What Application Performance Management Is

.....

In This Chapter

- ▶ Recognizing trends and challenges
 - ▶ Looking at different APM components
 - ▶ Bringing it all together in a complete solution
-

Today's complex business applications present new challenges that legacy application monitoring tools are ill-equipped to address. This chapter talks about these challenges and the important elements that make up a complete and effective application performance management (APM) solution.

Complex Application and Infrastructure Landscape

Increasingly, application performance directly impacts business performance. Users rely on applications to reach customers, build and sell products, provide

services, automate business processes, and perform almost every other task critical to the business. And as applications have become more critical, they have also become more complex. For IT teams, application performance and availability are the most visible indicators of their success.

Accordingly, IT organizations are changing the way they manage performance to be more holistic and application focused. In a 2012 study, Enterprise Management Associates found that 81 percent of participants regularly leverage cross-domain teams to tackle and manage application performance. Also in 2012, Gartner found that 20 percent of Global 2000 enterprises are reengineering their entire IT operational process framework in a way that shifts the central focus to the monitoring and management of applications.

The demand for newer, more powerful business applications at an ever-increasing rate has led to the adoption of rapid software development methodologies that increase the speed of delivery for new applications. At the same time, the introduction of new IT infrastructure technologies has increased the complexity of the underlying infrastructure over which applications run (see Figure 1-1). Some examples of these trends include

- ✔ **Agile development and DevOps.** Agile development is a conceptual software development framework in which cross-functional teams share knowledge, ideas, opinions, and experiences to identify evolving requirements and solutions. Team tasks are divided into short timebox iterations that typically last only a few weeks and cover all development functions including

planning, requirements analysis, design, coding, and unit and acceptance testing. DevOps is a rapid software test and deployment method that emphasizes communication and collaboration between development and IT operations teams.

- ✔ **Service-oriented architecture (SOA).** A software design methodology based on functional modules, or “services,” that provide the complete functionality of a large or complex, often web- or HTML5-based, software application.
- ✔ **Virtualization and cloud.** Virtualization and cloud technologies provide flexibility, rapid provisioning, operational efficiencies, and many other benefits, but they also introduce new server, storage, and network infrastructure complexities.
- ✔ **Edge computing.** Multi-tiered and highly integrated architectures mean that a portion of the application runs at the “edge” — closest to the user. Of course, when an application has an issue, it’s the experience at the edge that matters most to the user. But troubleshooting an issue in such a complex environment requires visibility everywhere that the application and its individual components run, including dependent systems, applications, and services.
- ✔ **Mobility.** In traditional networks, business applications run entirely on systems in the corporate network, thus IT can have complete end-to-end visibility into these applications and their performance. Increasingly, users are accessing these systems and applications beyond the network — from anywhere, at any time, and on any device.

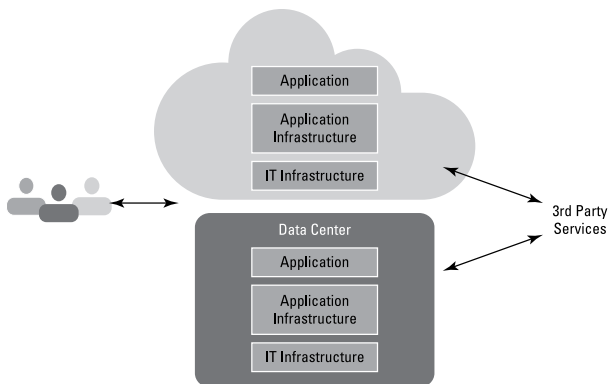


Figure 1-1: The complex world of the modern application.

In the face of these trends, IT and application teams must address challenges that include

- ✔ **Troubleshooting and recovering from business disrupting application issues.** Downtime and poor performance of critical applications have a direct impact on business. When application issues occur, IT operations and application and network teams must detect, isolate, and fix the problem quickly. But with legacy performance management tools, it can take days — if not weeks — to isolate and fix performance problems, and problems are often only discovered after end users report an issue.
- ✔ **Failing to find and fix recurring or persistent performance problems.** Intermittent and chronic performance problems negatively impact end-user

satisfaction and productivity and prevent IT from focusing on new initiatives. Although users often become numb to these chronic problems, they can still have a significant impact on business performance.

- ✔ **Preventing future disruptions due to an inability to detect problems early.** IT usually finds out about performance problems from end users (according to Gartner, up to 70 percent of the time), after the business has already been impacted. IT needs to detect problems proactively so that they are working on isolating and fixing the problem before the business is impacted.
- ✔ **Ensuring that IT initiatives don't cause new application performance issues.** When businesses virtualize, consolidate, or migrate their data centers to the cloud, they expect to improve flexibility, cost, and control. They don't expect to negatively impact application performance. When rolling out new applications or expanding existing deployments, it's critical to ensure that the performance required by the business will be delivered. Without the right tools to manage and predict the effects of such infrastructure or application changes on application performance, however, businesses often find themselves dealing with unforeseen performance problems.
- ✔ **Reporting pertinent information to executives and other key groups.** Application performance isn't a single group's responsibility and has broad implications across IT operations, application teams, and business owners. It's imperative in this environment of heightened application focus that

IT be able to communicate about application performance broadly and in languages tailored to technical and business audiences.

These challenges result in significant negative impacts to businesses, including loss of revenue and productivity, customer dissatisfaction, and frustration with IT.

Different Elements of APM

Today's complex application and infrastructure landscape requires an APM solution that looks across all critical parts of the application delivery chain. Whether you are rolling out new applications, consolidating or virtualizing data centers, or migrating to the cloud, managing application performance requires a holistic view that includes

- ✔ **End-user experience.** What is the end user actually experiencing as they interact with the application? Whether using web- or thick-client applications, and regardless of whether users are local, around the world, or mobile, you must be able to monitor and troubleshoot the ultimate measure of application performance: the end-user's experience.
- ✔ **Transaction tracing and component monitoring.** What an end user perceives as a single operation involves many distinct sub transactions across many different application components, such as different databases. How do you track, monitor, and troubleshoot the performance of each of these critical transactions, correlate them, and deep dive into the various components of the application?

✓ **Infrastructure and network performance management.** How is the performance and availability of the underlying IT infrastructure and network contributing to the performance of the application? IT infrastructure exists for one reason: to deliver applications. To really understand holistic application performance, infrastructure must be managed through the lens of the application. In particular, the network is a critical component whose importance continues to grow with cloud, mobile, SDN (software-defined networking), and network virtualization trends.



Although transaction tracing in APM tends to refer to code-level tracing within the servers, the network provides another critical perspective using packet captures to analyze network traffic. Both approaches have value and should be part of your solution.



SDN is a still-emerging approach to building computer networks that simplifies networking by replacing many of the functions of traditional network equipment (such as firewalls, routers, and switches) with software components.

Monitoring and Troubleshooting Applications

An effective approach to APM must integrate all APM elements. The complex, rapidly changing, and business-critical nature of modern applications demand it. A comprehensive APM solution enables IT teams to:

✓ Rapidly diagnose the root cause of performance problems.

- Identify problems fast by seeing all components of application delivery including IT infrastructure and the network.
- Identify performance problems sooner and avoid negative impacts to the business with advanced analytics that detect, isolate, and pinpoint performance problems.
- Eliminate the “war room” that takes IT resources away from more strategic projects and reduce or eliminate “finger pointing.”

✓ Dramatically improve IT efficiency.

- Streamline communication among IT teams (faster development life cycles, lower support costs, and fewer project failures).
- Leverage dashboards and reporting that present relevant information to both business and IT stakeholders.
- Understand how applications are performing with easy-to-digest performance snapshots.

✓ Proactively manage performance problems.

- Map the landscape of application dependencies (known as “application dependency mapping”) across all IT assets to perform impact analysis and to ensure all critical parts of the application delivery chain are monitored.
- Enable faster, more accurate planning to minimize the impact of IT change initiatives.

APM and ADC: Diagnosis and cure

Imagine going to your doctor because you have a sore throat. Your doctor sticks a tongue depressor in your mouth, tells you to say “ahhhh,” confirms that you have a sore throat, and sends you on your way!

This example is much like legacy application monitoring tools today. These tools provide a very focused, but limited, view of the overall problem. To effectively diagnose a problem in today’s complex application and infrastructure landscape, you need to see the complete picture.

Now, returning to the doctor’s office scenario, you’re more likely to have a nurse first ask you about your symptoms, medical history, and any medications that you’re taking, then check your blood pressure, temperature, and pulse. The doctor then gives you a complete examination, checking not only your throat but also your breathing, heartbeat, and ears, before making a diagnosis and prescribing an antibiotic.

This is more analogous to a complete APM solution — except for the prescription. If APM provides the diagnosis, then Application Delivery Controllers (ADCs), as well as WAN accelerators, provide the cure. Going a step further with this analogy, ADCs are also an integral part of your wellness program for preventative care. Thus, APM and ADCs work together to keep your applications environment healthy!

Chapter 2

From the Driver's Seat: End-User Experience Monitoring

In This Chapter

- ▶ Getting to know your end users
 - ▶ Measuring real-time performance
 - ▶ Monitoring availability with synthetic tools
-

Quality of experience (QoE) refers to the quality of the interaction between the user and the application from the user's perspective. While this can relate to all aspects of the interaction, including ease of use, intuitiveness, and many other facets, end-user experience (EUE) monitoring within application performance management (APM) focuses on the performance-related aspects of the experience, such as response time and availability.

In this chapter, you explore APM tools that provide insight into application performance from the perspective of your users. In other words, the view from the driver's seat!

Understanding User Behaviors and Patterns

In the not-too-distant past, availability (whether “up” or “down”) was the key measure of performance of networks, systems, and applications for IT teams. And while availability still matters, networking and systems hardware fail less often, and critical infrastructure now regularly incorporates redundant components and designs for high availability.

Today, end users expect more. Not only do applications have to be available where and when they’re needed — and on an ever-increasing number and type of devices — but they also must be responsive and consistently deliver a fast experience. End users and business units judge applications by their interactions and experiences with those applications. And while errors and downtime can still cause issues, speed or more precisely, lack of speed, measured in response time can be a major source of frustration for your users. Think about it, when was the last time your users complained that their applications were too fast?!

Performance has historically been measured at an individual component or system level, such as a network segment, a database, or an application server. The assumption was that if all the “parts” were performing as expected, then the user’s experience must be fine. But the sum of the individual parts often did not accurately represent the whole experience. Individual back-end measures, such as bandwidth and server utilization, provide a myopic view of application performance. What

really matters is end-user experience (EUE). Gartner defines EUE monitoring as “the capture of data about how end-to-end application availability, latency, execution correctness, and quality appear to the end user” (see Figure 2-1).

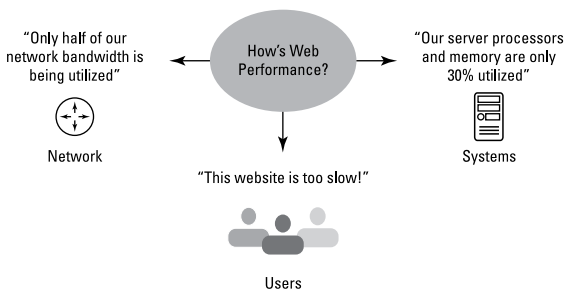


Figure 2-1: Different perspectives on performance.

EUE monitoring also contributes to a broader understanding of user behaviors and patterns with respect to applications. By monitoring real user interactions with applications, IT teams see how their users *actually* use an application, rather than how they are *expected* to use an application. This understanding not only improves troubleshooting and problem resolution, but it also can lead to business process improvement that is driven by IT! Thus, IT teams can transform themselves from a user and application support function that is reactive in nature, into a more proactive service-provider role that delivers added value to the business.

End-User Experience Monitoring

Analysis of EUE data enables IT teams to break down application response times into contributing sources, launch troubleshooting of root causes, and analyze traffic and response times among servers to manage multi-tier applications. EUE monitoring in application performance monitoring takes two different approaches: Real EUE monitoring (also known as Real User Monitoring, or RUM) and synthetic end-user monitoring.

Real end-user experience monitoring

Real EUE monitoring observes and records performance data about actual live user transactions within applications to determine whether these applications are meeting the users' performance expectations, as defined in service-level agreements (SLAs).



Effective end-user monitoring measures an application's responsiveness the same way that your users experience it, thus requiring a holistic view of the application. For example, web-transaction response time should be measured at the page level rather than being based on a single HTTP request/response performance measurement.

To monitor real EUE, either an agentless, appliance-based (physical or virtual) solution is installed, or agents are deployed throughout the application infrastructure. JavaScript-based EUE measurements, both real EUE monitoring and synthetic end-user monitoring (discussed in the next section), within web browsers can also feed performance data about web-based applications to a management server.



A passive appliance-based real EUE monitoring solution can provide additional capabilities for an IT team, including the ability to:

- ✓ Visualize communication within the data center and across the WAN
- ✓ Measure traffic by application, user, business division, and location
- ✓ Analyze historical information for trending and capacity planning
- ✓ Measure performance of voice-over-IP (VoIP) calls while they are in progress

Real EUE monitoring enables IT teams to:

- ✓ Measure application performance for internal and external users of your enterprise applications
- ✓ Monitor live performance for all users at all times, and alert against SLAs



JavaScript-based agents are a great solution for EUE monitoring of cloud-based applications, where deploying an appliance isn't an option.

Synthetic end-user monitoring

Synthetic end-user monitoring augments real end-user monitoring. With an understanding of your users' behaviors and patterns (explained earlier in this chapter), you can create a series of process steps within an application that run at a continuous or timed interval.

Synthetic end-user monitoring provides valuable information about an application's availability.



Synthetic end-user monitoring is only as good as its steps are accurate and relevant. You must accurately replicate the aggregate behavior of your users and the steps being monitored must indicate the availability of the application.

Chapter 3

Looking Under the Hood: Transaction Tracing

In This Chapter

- ▶ Triaging application issues with APM
 - ▶ Decoding application performance issues
 - ▶ Collecting and analyzing transaction data
-

What an end user perceives as a single operation actually involves many distinct application transactions across many different application components including the backend database, web server, middleware, auditing servers, and backend third-party services.

How do you monitor and troubleshoot the performance of every component of multi-tier applications and the transactions that traverse them? Roll up your sleeves and find out how in this chapter!

Following a Single Application Transaction Across All Components

Today's complex multi-tier business applications introduce new challenges for effectively triaging application-performance issues. These challenges include

- ✓ Monitoring applications across all tiers and domains, and understanding how transactions traverse application components
- ✓ Locating problems during development and in production, and rapidly pinpointing the source of performance issues at the component/code level



Transactions are a unit of work performed by an application. Most applications support many different transactions related to a common business area and depend on a common set of data. Each transaction is designed to accomplish a particular business purpose and exposes parameters to specify the particularities of the transaction each time it is requested by a user.

Monitoring Performance at the Code Level at Every Step

Incomplete data leads to incorrect answers. Therefore, an APM solution needs to collect data about the

performance of application components at a very granular level of detail, monitoring performance metrics across application components such as Java, .NET, web servers, databases, operating systems, and storage.

APM solutions need to provide the greatest visibility with the lowest overhead, which can only be accomplished with automated intelligence. It is impractical to think that IT administrators can manually determine what individual components should be instrumented.

Tracing Transactions

Application code-level transaction tracing bridges the workflow between application support and development, enabling cross-functional IT teams to identify and troubleshoot performance issues from the specific transaction step to the affected code. Examples of developer pain points related to application performance include

✓ Before deployment

- Performance results are too hard to get while performing quality assurance (QA) on code.
- Legacy code isn't instrumented (see Chapter 4 to learn more about instrumentation).
- Performance test results aren't actionable.
- Performance bottlenecks take days to find — sifting through load test results is difficult.

✓ In production

- Code is too often “guilty until proven innocent,” and developers have little or no access to production.
- Issues can’t be diagnosed to a level that tells developers what code needs to be fixed.
- Production problems can’t be easily or accurately reproduced.
- Architectures are becoming more complex, so finding problems becomes more difficult.

To address these pain points, transaction tracing enables application developers to see exactly what happened in any transaction across all tiers to

- ✓ Determine what application components come into play as the transaction traverses them
- ✓ Understand the impact of individual instruction sets on transaction performance
- ✓ Identify any external system dependencies that may exist
- ✓ Obtain visibility into the execution of third-party code and components
- ✓ Drill down from the transaction to the Integrated Development Environment tool (IDE) and the specific problematic application code

Chapter 4

Horsepower and Fuel Efficiency

.....

In This Chapter

- ▶ Collecting performance and forensic data
 - ▶ Going big with big data techniques
 - ▶ Automating analysis to detect performance problems
-

In this chapter, it's time to understand what powers APM — which is data. Here, you learn how application performance data is collected and how to get the most mileage out of this data!

Instrumentation: Keeping It Light

Instrumentation is a challenge for many APM solutions, particularly for multi-tiered applications with many application components. Other APM instrumentation challenges include mobility, virtualization, and cloud computing trends.

Instrumentation provides data feeds that enable an APM solution to determine the state of an application. Instrumentation types include

- ✓ Java or .NET code
- ✓ Installed agents (including databases, web servers, and middleware)
- ✓ Passive monitoring (agentless) via network-based, real-time packet inspection and analysis

Different deployment scenarios for APM instrumentation may include

- ✓ Agents on desktops and servers to continuously collect detailed application and systems data
- ✓ High-speed, passive appliances to measure end-user response time at key locations
- ✓ Lightweight JavaScript automatically downloaded to browsers to capture end-user experience for web-based applications
- ✓ High-speed passive appliances to monitor application traffic within the data center, across both physical and virtual infrastructure and servers
- ✓ Low-cost virtual appliances to measure user response time at branch offices and remote locations

Dealing with Lots of Data — Every Transaction

APM solutions capture copious amounts of data, including performance data and forensic data.

Performance data consists of measurements that are captured and recorded over time, such as CPU utilization, or the time it takes for a database query to execute. Performance data is monitored in real time for problem detection, and archived to provide historical views and analysis of trends. Anomalies that indicate a problem can be obscured by traditional data summarization techniques or missed completely by collection intervals that are too infrequent. An effective APM solution captures very fine-grained performance information and preserves its fidelity.

Forensic data goes beyond quantitative metrics and is obtained by recording activity, such as a transaction. Its purpose is to provide evidence or clues about a problem or operation (hence the term “forensics”) in the retrospective analysis of an event.

Forensic information is generally obtained from lower-level techniques, such as code-level tracing, detailed resource tracking (for example, memory consumption), and deep packet inspection of network traffic. The data is used to drill down deep into the behavior of the application, determine how components are interacting with each other, and break down all activity related to an individual transaction or other events of interests. Rich forensic information is fundamental to root cause isolation.

Applying Big Data Techniques

Big data techniques offer deep insight into application dynamics, thereby speeding problem diagnosis, increasing IT efficiency, and improving application service levels.

The full potential of APM lies in exploiting large volumes of detailed data about transaction performance and behavior. Big data capabilities go far beyond traditional statistical performance summaries by enabling massive retention of transactional data. These capabilities also enable powerful unstructured searches that give IT the ability to quickly detect patterns, and to find and analyze the specific transactions that definitively answer questions. Equipped with these capabilities, IT organizations can rapidly pinpoint the root causes of performance problems and minimize the impact on end users and the business.

The scale of data required to comprehensively manage application performance quickly overwhelms traditional performance management solutions. Virtualized, dynamic infrastructures and highly distributed application architectures further contribute to the exploding volume of data. Analytics are essential to pinpoint the specific data that is relevant to a problem, or to indicate an emerging problem. Historically, performance management solutions have dealt with this volume of data by sampling, averaging, or otherwise reducing the granularity of data they collect. These techniques leave IT teams with inadequate data and insight, losing the ability to track the complete application performance picture for each user and every application.

In contrast, big data techniques enable application support teams, developers, and IT operations to mine massive amounts of high-fidelity data to extract meaningful information. The result is faster troubleshooting, increased uptime, and the opportunity to proactively identify application issues before they impact business.

Leveraging Analytics to Detect Patterns and Problems

Analytics refers to a set of automated analyses that process and leverage performance and forensic data for performance management purposes, including monitoring, alerting, and troubleshooting. Once data is gathered, analytics operate on the data to provide functions such as:

- ✔ Adaptive baselining to establish “normal” behavior and uncover “abnormal” behavior to support dynamic thresholding and alerting
- ✔ Intelligent information storage for rapid retrieval during problem investigation or performance analysis
- ✔ Identification of metrics that are deviating from normal, based on complex variables such as time day/week/month, location, and user community, among others
- ✔ Correlation of performance anomalies to uncover cause-and-effect relationships, and patterns of deviation that occur together to determine if they are related and what they have in common (for example, a set of transactions) to provide a starting point for forensic analysis
- ✔ Derivation of new information to answer specific questions, such as long-term trends, peak operating periods, top consumers of resources, memory leak identification, and allocation of delay among application components

- ✓ Automation of complex administrative tasks, such as selectively instrumenting application components to maximize visibility while minimizing overhead



Gartner predicts that the importance of analytics to APM will grow over the next five years and recommends that IT operations analytics (ITOA) platforms be used to supplement APM suites.

Chapter 5

Exploring the Application Performance Management Terrain

.....

In This Chapter

- ▶ Ensuring satisfactory service levels
 - ▶ Diagnosing and assigning responsibility quickly
 - ▶ Troubleshooting performance issues methodically
 - ▶ Understanding complex application dependencies
 - ▶ Keeping an eye on applications in the cloud
 - ▶ Managing changes proactively
-

Application performance management (APM) gives IT teams visibility and actionable insight to deliver the application performance that users and businesses demand. In this chapter, you learn how APM helps IT teams manage service-level agreements (SLAs), perform triage quickly and effectively, troubleshoot problems, map application dependencies, and monitor applications in the cloud and through major infrastructure changes such as virtualization and consolidation.

Monitor Application Performance and SLA Compliance

APM enables IT teams to monitor SLA compliance beyond simple uptime measures. Although “five nines” (99.999 percent) availability remains an important SLA metric for service providers, end-to-end responsiveness has quickly become a more relevant bellwether of service levels for end users.

Although individual components in an application infrastructure may be running within established performance thresholds, the end-to-end user experience may be slow or unresponsive, making the application or website appear, in effect, unavailable or “down.” For example, the individual transactions of a website application will typically rely on the efficient operation of multiple steps and components such as:

- ✔ Web servers and services
- ✔ Application servers and services
- ✔ Database servers
- ✔ Physical network connections
- ✔ Routers and switches
- ✔ Firewalls and load balancers
- ✔ DNS (Domain Name Service) lookups

A failure or delay in any one of these steps or components results in the end user not being able to use the application. Additionally, although each of the individual steps or components may be performing as expected, the resulting performance may be unsatisfactory. Therefore, an APM solution must address end-to-end performance.

Perform Triage

When a critical business application is down or performing poorly, IT teams often lose valuable time pointing fingers at other teams (such as development, networking, and systems) or third-party vendors. Although such “blamestorming” may seem petty, it is often actually the result of logical, albeit flawed, deductive reasoning based on incomplete information from one-dimensional monitoring tools. The systems team sees that all servers are up and running optimally; the networking team reports that bandwidth utilization is nowhere near peak and latency is low; and the development team says that the custom application worked fine yesterday and no recent code changes have been introduced.

A comprehensive APM solution provides the entire team with a “big picture” view of the entire application and infrastructure landscape from end-to-end, including all transactions and components. IT teams can then drill down into the details of the problem and quickly triage and assign the problem to the appropriate team or third-party vendor.

Troubleshoot Application Performance

Troubleshooting an application performance issue is often a tedious and frustrating process that can take days or even weeks to resolve. The troubleshooting process is often further complicated by a lack of meaningful information about the problem from end users and limited tools for diagnosis and analysis.

A recent Forrester study found that only 25 percent of availability and performance problems is satisfactorily diagnosed and resolved within 24 hours. This represents a huge delta between reality and the SLAs by which IT organizations are measured (discussed earlier in this chapter), which often requires resolution of a critical problem within two hours. It is also a costly delta: the same Forrester study reports that more than 50 percent of respondents say the business cost of one hour of brownout (slow responsiveness) or downtime of their most critical application would cost from \$100,000 to more than \$1 million.



APM provides IT teams with the tools needed to effectively and methodically troubleshoot vexing performance problems, beginning with a “big picture” top-down approach that enables them to pinpoint and then dive deep into the individual transactions and components to isolate and resolve the root cause of performance issues.

Map Application Dependencies

Mapping application dependencies is an important process, particularly for complex, multi-tier applications. Such applications often have unexpected dependencies that even the application owners themselves may not understand. An incomplete understanding of business-critical applications makes troubleshooting difficult and increases the risk associated with application and infrastructure changes, such as cloud migrations and virtualization initiatives (discussed later in this chapter).

APM provides the capability to map an application to the application components and underlying infrastructure elements that deliver it. This understanding of the application architecture enables more effective troubleshooting and helps to reduce unknown risks associated with application and infrastructure changes.

Manage Application Performance in the Cloud

Cloud computing provides opportunities for greater operational efficiency. The on-demand computing capabilities of a cloud can be more flexible and cost-effective—but several application management challenges aren't so easily vaporized.



Just because you've moved your applications to a cloud doesn't mean you don't have to keep your customers happy and your vendors honest! APM challenges in the cloud include

- ✓ Measuring end-user experience when it may not be possible to instrument in traditional ways, such as via a network-based appliance
- ✓ Monitoring application components that are executing in the cloud
- ✓ Troubleshooting performance issues when the extended IT team includes the cloud provider
- ✓ Dealing with the complexity that is characteristic of hybrid environments that encompass multiple cloud vendors, as well as in-house infrastructure and applications

While cloud computing holds significant promise and claims to address many of the traditional challenges of application performance management, in reality, an organization's transition to the cloud will be gradual. There will always be a mix of application deployment models, and it would be wise to employ a management system that can handle them all. Important characteristics of such an APM solution include

- ✔ Measures the key aspects of application performance, including user experience.
- ✔ Provides a unified view in a hybrid environment comprising multiple cloud services and data center applications.
- ✔ Integrates application knowledge from multiple perspectives — including application components, network, server, database, and web services.
- ✔ Offers a holistic approach that treats the application as more than a sum of its parts, when possible.
Note: In a SaaS environment, it may not be possible to get an end-to-end view of the application.
- ✔ Includes real-time operational analytics.
- ✔ Provides appliance- and agent-based instrumentation options for a cost-effective approach to monitoring end-user experience.
- ✔ Fosters collaboration among IT teams and with cloud vendors.

Manage Performance with Application and Infrastructure Change

As applications and infrastructures change, critical business applications must be monitored to ensure that these changes don't unexpectedly impact performance.

Server virtualization and consolidation are key trends that are revolutionizing the modern data center. In many organizations, virtualization and consolidation initiatives are primarily driven by financial considerations that include reducing management and administrative costs.

However, application performance is often an afterthought. This can be a costly mistake as virtualization inherently reduces visibility and increases application complexity. To proactively address these performance issues, virtualization initiatives should include implementation of comprehensive APM capabilities. Without adequate visibility, these types of projects may be significantly delayed, eliminating or reducing potential cost savings.

Chapter 6

Ten (Okay, Five) Application Performance Management Evaluation Criteria

In This Chapter

- ▶ Watching the end-user experience
 - ▶ Diagnosing performance problems
 - ▶ Mapping out application dependency
 - ▶ Talking about analytics
 - ▶ Looking at dashboards
-

This chapter reviews several of the important capabilities of APM and provides convenient checklists to help you evaluate different APM solutions.

End-User Experience (EUE) Monitoring

End users expect instant access and a consistent experience with the business applications they need. You

must be able to monitor and troubleshoot the ultimate measure of application performance: the end-user's experience. Use Table 6-1 to compare APM solutions.

Table 6-1 EUE Monitoring Checklist

<i>Criteria</i>	<i>Yes</i>	<i>Partial</i>	<i>No</i>
Monitors live performance of real users continuously			
Measures web transaction response at the page level, the same way users see it			
Establishes a baseline and generates alerts for poor end-user experience			
Provides complete visibility into end-user geography, platforms, and usage trends			
Determines the share of end-to-end delay that is attributable to each of the major components (for example, network, server, client, and others)			
Monitors the availability of applications even when there is no end-user traffic, using synthetic transactions			
Provides visibility in EUE for specific locations, applications, and other business-oriented criteria			

Transaction Tracing and Application Component Monitoring

Code-level transaction tracing and application component performance monitoring enable you to see deep inside your applications to diagnose the root causes of performance problems. Table 6-2 lists several important criteria.

Table 6-2 Transaction Tracing Checklist

<i>Criteria</i>	<i>Yes</i>	<i>Partial</i>	<i>No</i>
Traces user transactions across all application tiers			
Records and indexes all transactions, not just samples			
Monitors performance data at a fine-grained level to ensure that anomalies are captured			
Supports multiple application environments, including Java, .NET, popular third-party software, and custom applications			
Monitors system and application metrics			
Has very low operating overhead			
Configuration is simple/ automated			
Uses a highly-scalable, big data approach to analyze transactions			

Application Dependency Mapping

A run-time application dependency map provides a complete picture of your application and infrastructure components, and their relationships. See what's important in Table 6-3.

Table 6-3 **Application Dependency Mapping Checklist**

<i>Criteria</i>	<i>Yes</i>	<i>Partial</i>	<i>No</i>
Produces an accurate map of application dependencies at the time of execution			
Maps dynamic operating environments, such as SOA and virtualized servers			
Leverages auto-discovery or existing data, and does not require manual data input or maintenance			
Finds and maps all applications, including custom developed applications			

Analytics

Analytics monitor system and application metrics to automatically detect anomalous behavior, correlate information, identify the root cause of problems, and predict events and performance trends. Refer to Table 6-4 to assess APM strengths in analytics.

Table 6-4

Analytics Checklist

<i>Criteria</i>	<i>Yes</i>	<i>Partial</i>	<i>No</i>
Detects and correlates performance anomalies			
Captures and indexes every user transaction			
Extracts answers from vast amounts of performance and forensic data			
Utilizes a natural search language capability to quickly pinpoint meaningful information			

Dashboards

Dashboards provide a big picture view of your applications and infrastructure in a “single pane of glass” that provides different teams with actionable alerts and information. See what’s important in Table 6-5.

Table 6-5 Dashboards Checklist

<i>Criteria</i>	<i>Yes</i>	<i>Partial</i>	<i>No</i>
Allows fully customizable views and role-based access controls for individual users or user groups			
Provides high-level views and alerts for fast triage and diagnosis			
Facilitates root cause analysis with seamless drill-downs			
Enables creation of new dashboards using drag-and-drop workflows			

Notes



Notes

.....

KISS

PERFORMANCE PROBLEMS GOODBYE
BEFORE THEY SAY HELLO.



What if you could streamline performance management and bring together the complete application picture including end user, application transactions, IT and network infrastructure? And what if you could apply “big data” techniques to automate analysis and troubleshooting, so you can solve problems in minutes or hours, instead of days or weeks? These “what ifs” can become reality with one introduction. Meet Riverbed.

riverbed Think fast.®

riverbed.com/kiss

©2013 Riverbed Technology. All rights reserved.

These materials are the copyright of John Wiley & Sons, Inc. and any dissemination, distribution, or unauthorized use is strictly prohibited.

Application performance is business performance

Application performance management helps deliver the application performance users and the business demand. Effectively monitoring and troubleshooting application performance issues requires a comprehensive solution. In this book, you find out how!

- *Understand real end-user experience monitoring — and why it's important*
- *Apply big data techniques and analytics — to quickly triage problems and minimize downtime*
- *Evaluate performance management criteria — and choose the best solution to meet your needs*



**Open the book
and find:**

- How software trends affect APM
- Why end-user experience monitoring is a key performance metric
- How transaction tracing enables troubleshooting at the code level
- How application dependencies across multi-tier components affect performance

Making Everything Easier!™

Go to [Dummies.com](https://www.dummies.com)[®]
for videos, step-by-step examples,
how-to articles, or to shop!

FOR
DUMMIES[®]
A Wiley Brand

ISBN: 978-1-118-71965-7
Not for resale

These materials are the copyright of John Wiley & Sons, Inc. and any dissemination, distribution, or unauthorized use is strictly prohibited.