

# Application Performance Starts with Database Performance Analysis

## About the Pie Chart

The data presented in the chart is derived from the 2014 Gleanster survey on Application Performance Management (n=182). The data presented in the body of this report reflects the findings from this survey.

The data serves as the basis for this Gleansight Deep Dive, which provides analyst commentary related to a particular aspect of the topic. The objective is to provide additional perspective and illuminate certain key considerations regarding the implementation of the related technology-enabled business initiative.

*To learn more about Gleanster's research methodology, please click here or email [research@gleanster.com](mailto:research@gleanster.com).*

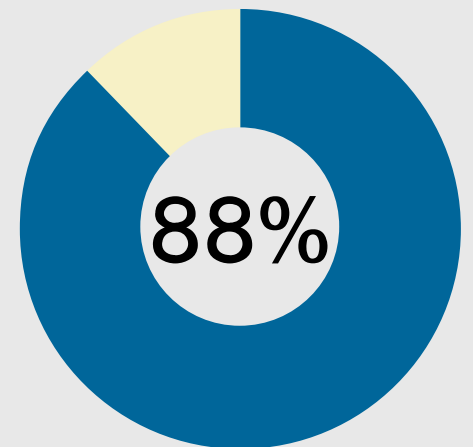
## Executive Summary

Application performance management (APM) tools have achieved impressive levels of adoption by providing a standard, automated method to identify application-level performance issues. Pervasive APM tools have helped create a new culture of continuous performance monitoring and improvement, but many of these tools treat the database as a black box simply executing queries. Yet respondents in the 2014 Application Performance Management survey (n = 182) reported that the number one source of issues with application performance is the database. In fact, 71% of respondents indicated their current application performance management tools provide hints, but rarely identify the root of problems.

This Deep Dive explores the disconnect between traditional APM tools and the reality that most production performance issues are firmly rooted in configuration or schema-related problems with the relational databases that still run a large majority of today's mission-critical enterprise applications. During this analysis a subcategory of APM is introduced – application database performance management.

## PERCENTAGE OF ALL RESPONDENTS

who cite the database as the most common challenge or issue with application performance.



ANCHORING STAT

**Note:** This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

### Top Performers Defined

Gleanster uses 2-3 key performance indicators (KPIs) to distinguish “Top Performers” from all other companies (“Everyone Else”) within a given data set, thereby establishing a basis for benchmarking best practices. By definition, Top Performers are comprised of the top quartile of qualified survey respondents (QSRs).

The KPIs used for distinguishing Top Performers focus on performance metrics that speak to year-over-year improvement in relevant, measurable areas. Not all KPIs are weighted equally.

The KPIs used to distinguish Top Performers in this Deep Dive include:

- Revenue growth

### Key Takeaways:

- 71% of respondents indicated their current application performance management tools provide hints, but rarely identify the root of problems.
- 76% of respondents indicate manual gathering and analysis of log data is frequently necessary to diagnose the issue.
- Seventy eight percent (78%) of Top Performers also stated that critical database performance indicators are tracked alongside application performance metrics.
- Eighty four percent (84%) of respondents reported investing in database monitoring tools, with a majority reporting that they rely on DBAs to perform performance monitoring
- This Deep Dive defines a subcategory of APM – application/database performance management (ADPM). This subcategory covers tools that incorporate information not just about queries being executed against a database by a particular application, but information about execution plans, database configuration, storage performance, index usage and contention, and other factors that are frequently identified as the root cause of application performance issues in databases.

With the current emphasis on horizontal scalability and cloud-based deployments, application developers and system administrators demand systems that give them a view into application internals. Application performance management (APM) solutions provide such a window by offering an instrumented view into an application’s internals, allowing architects, administrators, and developers to gain valuable clues about the root causes of performance issues. The current iteration of APM provides only part of the answer, with 62% of organizations supporting three or more APM solutions. While APM solutions provide valuable insights into the symptoms of performance problems, over 90% of these tools focus exclusively on application-level issues while ignoring database and infrastructure-related issues, which are often at the root of performance problems. APM tools aim to provide a unified picture of overall application performance, yet 76% of respondents

indicate manual gathering and analysis of log data is frequently necessary to diagnose the issue.

When a system can scale from hundreds to thousands of instances to meet instantaneous demand it is no longer enough to assess single-node performance. As such, a collection of application performance management tools has emerged to address the immediate needs of developers and administrators who are responsible for managing, monitoring, and reacting to performance problems that invariably occur in today’s critical systems.

A wide range of tools exist to address the specific area of application performance management, from New Relic, Compuware, to BMC, but only a small percentage of these tools analyze database performance alongside application activity. While APM tools have been created to instrument popular platforms such as PHP, Java, and .NET, most APM tools tend to leave database performance management to database

**Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.**

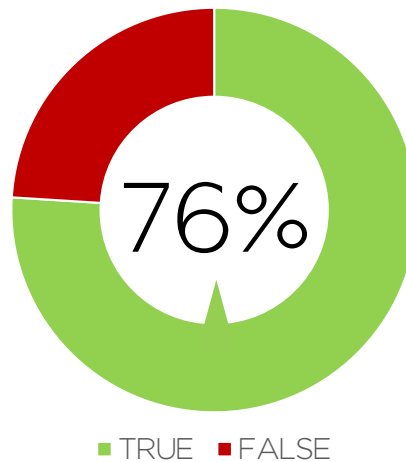
“...by monitoring the application and database simultaneously... Top Performers quickly identify where the bottlenecks exist -application, database, network, etc.”

vendors. In addition to the core APM tools designed to instrument and report on application performance, there are a variety of visualization and analytics tools designed to ingest and report on application data in real time. While the market gravitates toward several top players, the on-the-ground reality of application performance management is that many organizations are still forced to roll their own solutions. This Deep Dive explores one such gap in the current array of APM offerings – the lack of insight into database performance bottlenecks and the correlation of database performance issues with application performance management tools.

## Challenges in Application

### Figure 1: Continued Reliance on Manual Analysis

After a performance problem in production, teams are often manually gathering log data to diagnose the problems.



\* All Respondents, 2014 Gleanster APM Survey, n=182

## Development and Monitoring in 2014

Seventy two percent (72%) of Top Performers identified “close alignment of DBAs with application developers” as

a critical factor for success versus 31% of Everyone Else who claim the same. This suggests that Top Performers have learned from experience that there is little difference between the application layer and the underlying database. Seventy eight percent (78%) of Top Performers also stated that critical database performance indicators are tracked alongside application performance metrics, vs. 35% of Everyone Else who said application and database metrics are tracked separately.

Top Performers have achieved close alignment across application development teams and database administrators to achieve greater reliability, with 88% of Top Performers identifying the database and the network as the top two most important factors in production performance. The critical difference between Top Performers and Everyone Else is that Top Performers approach APM holistically by monitoring the application and database simultaneously. With this correlation, Top Performers quickly identify where the bottlenecks exist -application, database, network, etc.

Eighty four percent (84%) of respondents reported investing in database monitoring tools, with a majority reporting that they rely on DBAs to perform performance monitoring. Eighty eight percent (88%) of respondents stated that DBAs are the primary owners of database performance bottlenecks, identifying these database professionals as the first to respond to production performance issues. Based on Gleanster’s interviews and analysis with industry practitioners, we estimate database performance issues take on average two to three additional hours to address when compared with application issues. Where an application performance

Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

### Fast Fact

Eight out of ten survey respondents used application performance monitoring tools.

issue took an average of 1.5 hours to address, a production issue with a root cause in the database generally took over 3 hours. This variance is partly due to the fact that database issues tend to require escalation to specialized on-call database administrators to both identify and address the cause of an application issue. In today's always-on environment the difference between two hours of downtime and four hours of downtime isn't just measured in revenue lost, it is measured in customers lost.

This data illustrates the current disconnect between how organizations monitor and react to performance issues in the application and how they react to performance issues in the database. Top Performers admit the obvious – the application and the database equally as valuable when investigating issues.

## Application Performance Management Is Incident Response

Experienced industry practitioners understand that application performance management is essential when dealing with a performance crisis in a production system. When mission critical software is slow, unable to transact business, and when end-users are frustrated, this is when an APM tool is used to quickly diagnose problems. Today's sophisticated, multi-tier application architectures present dynamic challenges for architects and administrators as the size of IT portfolios continue to increase along with the number of technologies employed.

When a performance problem occurs in a mission-critical public-facing website these events often lead to headlines and headaches for end-users. Whether it is a multinational bank being unavailable for several days due to a database outage or a national electronics retailer

suffering an outage on Black Friday due to a spike in mobile traffic, performance problems are availability problems. While the case is clearly made for public-facing websites, the same can be said for internal applications. An organization can grind to a halt if a critical accounting system or inventory system suffers unexpected slowdowns, and as more businesses move to SaaS and DBaaS offerings in the cloud to support project management, customer service, and other business critical functions, performance is built into service level agreements with SaaS vendors.

### Trend: Application Performance Is the Critical Metric.

When a performance issue occurs in production it is often the equivalent of downtime – a performance problem in production is an emergency to be immediately addressed with the use of an APM tool. This emphasis on performance as a first-class incident in production is a relatively new trend, especially for enterprise software. A decade ago internal end-users using an accounting system or any other internal system couldn't reasonably expect performance on par with the consumer internet, where responses in the sub-500ms range make the difference between success or failure.

As more internal applications migrate to web-based interfaces and as more companies make use of software-as-a-service offerings such as Salesforce or Atlassian the expectation of immediately responsive applications has become the accepted norm. Users who demand immediate, 250 millisecond responses from Google have come to expect the same from all the software they use at work regardless of whether that software came from a billion dollar internet giant, an internal application development group, or an external SaaS

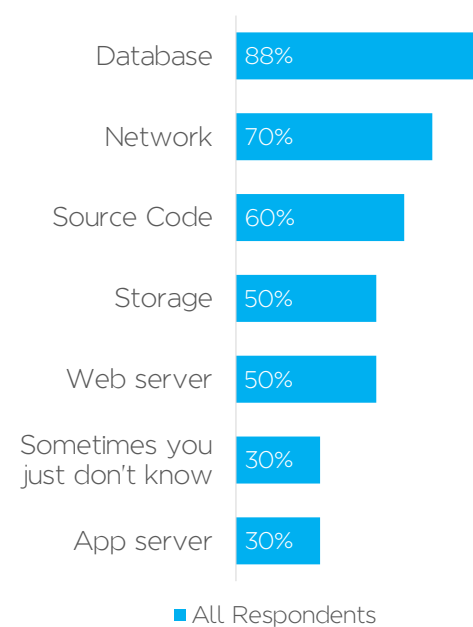
Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

vendor. In short, both internal users and consumers are now equally as impatient and less tolerant of an application that takes seconds to respond.

**Focus: APM Tools Lack Insight into Database Internals.** This Deep Dive defines a subcategory of APM – application/database performance management (ADPM). This subcategory covers tools that incorporate information not just about queries being executed against a database by a particular application, but information about execution plans, locking/blocking, storage performance, index usage and contention, and other factors that are frequently identified as the root cause of application performance issues in databases.

ADPM does away with the artificial

### Figure 2: Top 2 Reasons for Application Performance Issues



\* All Respondents, 2014 Gleanster APM Survey, n=182

separation between an application's data and the logic used to process this data in an effort to address performance issues related to the combination of query patterns driven by the application,

storage and query executions plans as implemented in a database. The root cause for many application performance issues is often a product of these two factors.

- A schema optimized for OLAP or reporting-style queries being accessed by an application that requires more OLTP or transaction-oriented behavior will perform poorly due to issues that are often only obvious to a DBA with the proper training and background required to identify these architectural disconnects.
- Another root cause for many unpredictable application performance issues is the complexity of query optimizers in dynamic systems. An execution plan for a common query can be changed in response to the size of tables, the order of clauses in a query, or changes to the configuration or storage systems that support a database.

These are just two examples of the kinds of nuanced and difficult to discover problems which come to characterize the debugging process for application performance issues. In many ways, ADPM tools aim to demystify database internals and democratize access to the arcane database performance information that is currently accessible only to experienced DBAs. In much the same way that APM tools make an organization less reliant on senior application developers for production debugging, ADPM tools allow organizations to operationalize the diagnostic functions currently performed by a DBA.

While most business applications are backed by a relational database, ADPM isn't necessarily limited to the analysis of relational database performance.

Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

**Remember**

What's it take to be a Top Performer? Top Performers were identified by isolating survey participants that actively used DAM to support marketing efforts AND achieved the top 25% highest self-reported year-over-year revenue growth. We use the aggregate survey trends to isolate what they do differently to support their success.

As most applications migrate to a more heterogeneous use of persistent storage, with key-value stores, HDFS, and distributed SQL engines having expanded the concept of the database, ADPM could also start to address this variety, fusing application performance metrics with database metrics for these alternative systems. The initial "frontier" for ADPM tools will be the reliable and ever-present relational database that continues to be the bedrock of modern-enterprise applications.

cover the database as an integrated component of an application, not as an external service used by the application. For many applications there is little difference between the application and the database it uses. This is the fundamental change that needs to happen to keep APM useful and relevant as organizations continue to invest in applications that ultimately make use of a relational database

**Application Database Performance Monitoring**

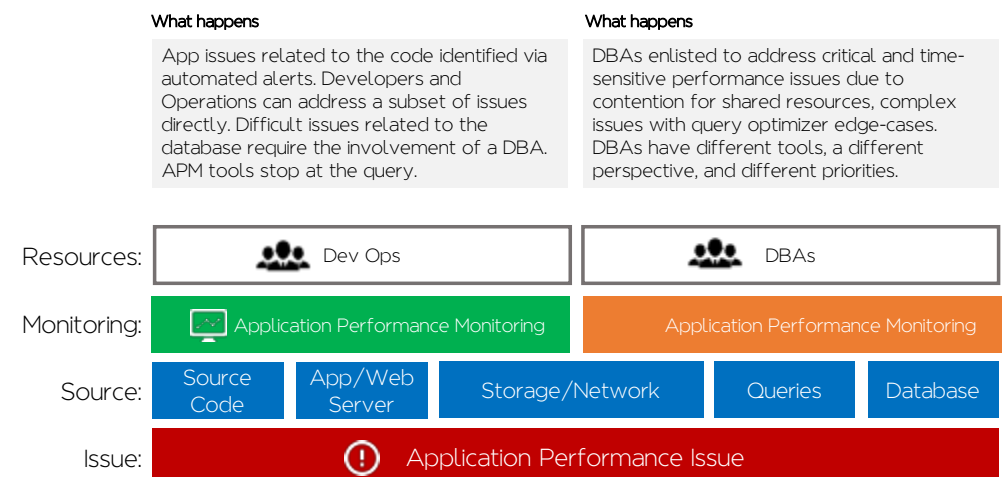
DBAs deserve some credit. While application developers ran toward APM, DBAs have been steadily measuring and monitoring database performance for years. The only problem is that this measurement and monitoring is almost always disconnected from the inner workings of the databases. Just as the relationship between application developers and database administrators is prone to miscommunication, so is the monitoring effort.

**Relational Databases: Still Central to Most Applications**

While there are organizations such as Facebook and Google that have built systems on massively scalable horizontal database technologies like Google Spanner, the vast majority of mission-critical applications still have a central dependency on a relational database. Even with the recent popularity of NoSQL databases such as Cassandra and MongoDB, along with a crowded field of other options, most systems using NoSQL databases tend to also make use of a relational database in some way. From SaaS products to eCommerce to Big Media, most websites and applications have at their core a large relational database

For this reason, application performance management needs to be extended to

**Figure 3: Bottlenecks in Investigating Application Performance Issues**



**Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.**

“Part of the problem of APM is ownership. While developers own the application stack, DBAs are often responsible for monitoring and mitigating problems with the databases.”

running in a standard database like Oracle, MySQL, SQL Server, DB2, or Sybase.

While the largest relational databases are sharded or running in a large cluster of machines, many of the most critical applications still rely on one or two instances of an RDBMS. These are the systems that have been running for years that are associated with DBAs who are used to managing these databases directly. Part of the problem of APM is ownership. While developers own the application stack, DBAs are often responsible for monitoring and mitigating problems with the databases, and many are skeptical of granting anyone (especially developers) access to statistics about the inner workings of the database.

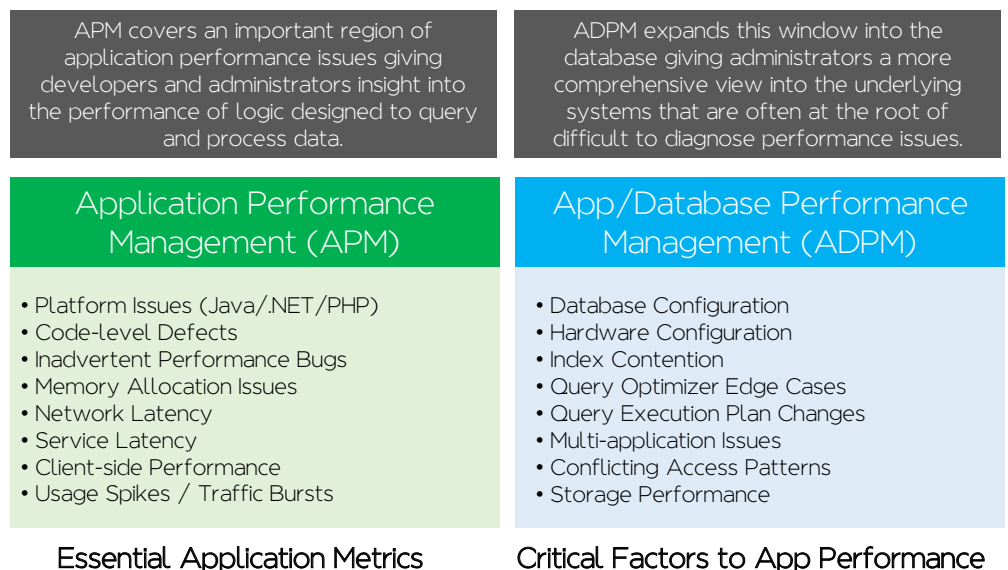
**The Folly of Application Monitoring: ‘Can someone call a DBA?’**

Experienced architects, developers, and administrators understand that the, bottleneck at scale is almost always the database and that these problems are often difficult and possibly intractable without the direct assistance of a DBA.

When an order processing system or an accounting system freezes due to system load at a critical time an APM tool that doesn’t understand the underlying database can only help so much. In these situations, when a system is unavailable due to contention on a table index, an unpredictable problem with an execution plan, or a problem with I/O in the database, the only reliable course of action is to call a DBA.

If application performance management is about addressing immediate performance emergencies in production, then they do a poor job at providing tools to monitor and manage the most common cause of instability at scale – the relational database. It is this inability to incorporate application-correlated database analysis with application performance management that points to an obvious gap in the conceptual model for APM. If the relational database still plays an important part in more enterprise applications then it is indistinguishable from the application and it needs to be monitored and instrumented alongside the application code that makes use of it.

**Figure 4: Expanding Performance Management to the Database**



Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

At scale and under heavy load APM tools start to break down. While they might point in the right direction, they won't uncover the details of an underlying database bottleneck. APM tools provide some level of detail on database queries, allowing users to track the most common queries and the queries that are taking the most time, but few APM tools dig deeper than the library or protocol used to communicate with the database. It is this failure to incorporate underlying database performance issues in APM that holds the entire market back. Instead of being able to use a single tool across the entire technology stack an organization is forced to "call the DBA" in the event of a performance problem in production.

### **Inaccurate Performance Management: A False Sense of Control**

As APM tools provide limited insight into database internals, they give architects, developers, and administrators a false sense of control, of confidence that they understand the factors affecting performance. The industry is full of examples of organizations that have suffered unavailability or performance issues at critical moments. In many cases these applications are overwhelmed by unexpected shifts in usage patterns, and for many systems the limited resource is almost always a relational database.

Current APM tools provide a valuable tool during development and pre-production testing, as application developers can use the data they provide during load testing and system certification to make continuous improvements to code execution. If a method or component is added to an application that increases a critical page load time, this method or component can be quickly modified to achieve

greater efficiency. What the current array of APM tools fail to address is how these applications use and place load on shared database resources in production. APM tools tend to generate a list of symptomatic issues without understanding the root cause of performance bottlenecks – hence the need for a redefinition of APM to include all aspects of system architecture: application database performance management (ADPM).

### **Benefits of Application Database Performance Management**

With an application data performance management (ADPM) tool those responsible for monitoring and mitigating the risk of performance bottlenecks will gain a more comprehensive perspective on how and where performance bottlenecks can occur. ADPM tools provide the following benefits over traditional APM tools:

#### ***Ability to correlate an application's queries with database performance.***

While an APM tool will stop at the query and treat the database as a black box, an ADPM tool dives into the internals, looking at index status and query execution plans to understand how a specific application is affecting the database. Without this level of instrumentation traditional APM tools will miss critical performance bottlenecks caused by unpredictable shifts in execution plans which occur when database configuration is modified, workloads change, or data growth increases.

#### ***Cross-application performance management for shared database resources.***

While an APM tool focuses on a single application running on an application server, an ADPM tool can assess the impact of multiple

Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.



“This traditional disconnect between application developers and database administrators is the result of years of tension in the role of the database in a production system.”

applications on a shared resource. This is often the single biggest contributor to performance bottlenecks in a database. When more than one application uses a shared database these applications can cause contention. With an ADPM tool DBAs and developers gain the ability to track how multiple applications are contributing to locking issues, index problems, and other difficult to manage database problems in a single, consolidated interface.

It is this single consolidated interface that drove the adoption of APM. APM tools rose to prominence as more and more organizations understood that standard application-level performance monitoring tools would lead to faster resolution of critical performance issues. As organizations start to incorporate APM tools into production monitoring systems the presence of an on-call DBA for every performance issue will become increasingly obvious.

With an ADPM tool organizations can start to automate and operationalize what once involved hands-on work from an advanced database professional. They can move quickly to identify and address performance problems as they occur.

## Building a Business Case for Application and Database Monitoring Tools

Ultimately an APM tool is in place for four reasons:

### 1.) ADPM Results in Cost Savings

APM tools reduce the time spent identifying issues. What was once an open-ended exploration into code and database-level issues is now a standard, “operationalized” aspect of a production deployment. While development and database specialists

are still required to write software, they do not need to be employed to identify the cause of a performance bottleneck.

With the expansion to ADPM to cover database monitoring, this cost savings is expanded to include not only the application code but also the database as the source of most performance outages. While it is common to enlist the help of a DBA in an open-ended exploration of database performance issues, with an ADPM tool an organization gains a more holistic view of performance that relates application code to the current state of the database.

### 2.) ADPM Mitigates Risk and Reduces Downtime

As a performance issue in production is not as critical an issue as application downtime, an APM tool reduces an organization’s risk. When a critical performance bottleneck occurs in production, limited resources can focus on finding a remedy rather than pointing fingers.

This traditional disconnect between application developers and database administrators is the result of years of tension in the role of the database in a production system. With an ADPM tool an organization can mitigate the risk that this disconnect will delay the identification of serious performance issues as developers and DBAs struggle to place the blame for database performance bottlenecks on each other

### 3.) ADPM Tools Support Change Management

While APM tools are rarely considered change management tools, they serve as an important historical record of changes and the impact of those changes on application performance. As performance is closely associated

Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

with availability and uptime, the ability to show management an audit of performance data over time is evidence that an IT department is tracking key performance indicators as new code is released to production.

With an ADPM tool, performance changes can be monitored as both code-level and database-level changes are introduced into a running system. With traditional APM tools stopping at the protocol level it is difficult to get a clear picture of what exact changes contribute to a performance bottleneck. With an ADPM tool administrators and architects can correlate performance bottlenecks to changes in the code, the schema, query execution plans, or database configuration

#### **4.) ADPM Tools Provide More Comprehensive Performance Monitoring**

As large organizations continue to transform into 24-hour operations supporting customers and employees around the world, having a tools to automate and mitigate performance bottlenecks as they happen is essential. As complex performance problem often require the most senior technical expertise in an IT department, anything that can help identify issues quickly will reduce load on critical resources, making it more likely that an organization's productive development resources can remain focused on creating value rather than on supporting production.

With an ADPM tool organizations are automating and operationalizing this technical expertise of identifying performance bottlenecks for an environment that is transitioning from an on-premise 8/5 reality to an always on, in the cloud, 24/7 reality. In today's fast-paced environment business can

no longer wait for critical resources to "jump on a call" to help diagnose performance issues. An ADPM tackles that last mile to help organizations track once-mysterious database performance metrics alongside an application.

### **Making ADPM a Priority**

Survey respondents from large organizations identified growth in the IT portfolio as a constant pressure on the organization. As IT departments support an increasing number of applications, an important differentiator for Top Performers is the ability to maintain operational IT support functions for production-facing systems. ADPM plays an important role in helping IT departments to remain productive and proactive, creating value for customers.

Business continuity and the need to adapt to a 24/7 operating environment also place a high burden on senior technical resources, who are often called to support tricky or difficult production performance issues. Without ADPM an organization must dedicate a DBA per application for production support. With ADPM an organization has full visibility into database performance metrics that is aligned with the usage patterns introduced by specific applications. In short, instead of having to constantly call a DBA to manually check the status of a database, with an ADPM you gain a better understanding of both database metrics and how database metrics align with specific applications.

### **ADPM Spend Results in Greater Productivity**

IT professionals understand that once an application is deployed to production the productivity for both developers and DBAs tends to decrease as systems are debugged and performance problems are diagnosed. This is a built-in part

Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.

## Questions to Ask about APM and ADPM

When evaluating whether an APM or ADPM is right for your organization, ask yourself the following questions. If you answer yes to any of these questions you should evaluate an application database performance management solution.

- Has your application suffered an outage or performance degradation in the last 12 months due to database performance issues?
- Are you overwhelmed by metrics in your current application monitoring tools, and are you looking for simple answers to questions about why performance is affected by changes to code, infrastructure, or configuration?
- Has an application's performance been affected by changes to the underlying database hardware or storage subsystems?
- What change control mechanisms are in place to help DBAs troubleshoot issues when developers deploy new code?
- Has your application experienced sudden degradations in performance due to query execution plans adapting to new usage patterns?
- Does your organization have multiple applications using the same shared database, and do applications compete for shared resources?
- Are you concerned about the customer impact of application performance problems?

of the software development lifecycle. When a new application is deployed to production, barring a rigorous load-testing effort for pre-production, there are almost always unanticipated performance issues and operational challenges that only exist in production.

No matter how much an organization prepares for production, there is always a surprise in store, and this often translates to senior technical resources being tied up in a support function for days, weeks, or months following a major software release. A comprehensive ADPM solution reduces this operational overhead. While senior technical resources will still fix critical bugs, with an ADPM it will take far less time to identify the cause of performance bottlenecks.

In this way an ADPM increases productivity in the IT department, allowing teams to develop and deliver software without having to set aside large blocks of time for DBAs and administrators to track down difficult issues related to execution query plans, table or row locks, and contention for database resources. While an APM tool forces you to make use of expensive, limited DBA resources, an ADPM reduces the need to involve a database specialist when identifying problem. This results in greater productivity, as resources previously occupied with operational concerns can focus on new application development. When economic buyers consider the cost of a DBA's time and the opportunity cost associated with development resources, augmenting these efforts with an ADPM offering can provide a substantial return on investment.

### The Blame Game Stops Today

To a database administrator it is always the application. There's nothing wrong with the database, it's just that those

application developers don't know how to use it correctly. To application developers it is always the database. The application doesn't do exactly what the customer asked us to do – if the database can't keep up then how is that the application's fault?

This is just one of the constant battles within the modern IT department: two roles supporting the same application with a completely different view of how that application uses a database. Using the APM tools allows developers to retreat to the application without bridging this disconnect. Using isolated database monitoring tools gives DBAs an excuse to throw their hands up and complain that application developers don't understand schema design or elegant query construction.

An ADPM tool properly employed gives neither side of this debate any shelter and forces teams of developers and DBAs to work together on a shared model of application and database performance. Instead of allowing IT departments to compartmentalize applications from databases, an ADPM tool puts everyone on the same team, showing application performance and database performance in the same place. This reduces the territorial nature of the Dev. vs. DBA conversation and results in less finger pointing. In many cases performance problem emerge due to miscommunication, and ADPM reduces the likelihood of just this kind of siloed thinking about performance.

### The Economic Case for ADPM

The economic case for ADPM vs. APM is straightforward. Take the risk introduced by downtime in a production-facing system. Major disruptions to an application may take 30 minutes to diagnose with an APM; if these disruptions involve a difficult-to-diagnose issue in a shared database it

may take another 30 minutes to locate the appropriate DBA and an additional 60 minutes to simply identify the root cause of a performance bottleneck (well, 60 minutes if you are lucky - root cause analysis can take weeks, months and even years). This is approximately two hours spent during the discovery phase of an incident response.

With an ADPM that integrates application-level calls with database-level calls you can skip the escalation process to a DBA because ADPM automates and operationalizes the identification process for database issues. Instead of having to call an on-call DBA to identify which index or which query plan is having problems, an ADPM relates a page load time or an application use case with a specific query and flags an issue with a database resource proactively.

The choice is simple:

- Organizations can employ an APM tool that reports on a symptom of a database performance bottleneck, spending additional hours to work with a database professional who will then run database-specific monitoring tools, OR
- Organizations can employ an ADPM tool that reports on end-to-end performance that correlates application behavior with specific database configuration.

In summary, this is a choice between monitoring the root causes of performance problems, or not. Any performance management tool that doesn't incorporate database monitoring is missing a significant portion of the overall performance landscape, as it is the database that presents performance issues at scale.

## Conclusion

Current approaches to APM are a

dramatic and impressive improvement over application performance management approaches of a decade ago, but current best-of-breed APM solutions still treat the database as a difficult-to-manage external resource. This disconnect between application performance monitoring and database performance monitoring leaves critical gaps in the systems that monitor application-level performance in production, leading to unnecessary time enlisting database professionals to track down the most common sources of performance issues in production systems.

To address this gap a new class of APM tools – application database performance management tools – aim to correlate application-level metrics with database-level metrics to provide administrators with a comprehensive overview of end-to-end application performance. As more organizations come to realize that application and database performance cannot be arbitrarily separated, APM tools will start to drift toward full-stack suites covering everything from the web server down to the status of partitions and indexes in the critical relational databases that continue to provide the foundation for enterprise applications.

**Note:** This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.



## Related Research

Recently published research that may be of interest to senior industry practitioners include:

The Right Data at the Right Time: Bringing Agility to Data Management

How Top Performers Improve Data Quality for Better Business Intelligence

2014 Digital Asset Management Benchmark

2014 Web Content Management Benchmark

5 Considerations Every Successful Web Content Management Initiative Must Address

The Gleanster website also features carefully vetted white papers on these and other topics as well as Success Stories that bring the research to life with real-world case studies. To download Gleanster content, or to view the future research agenda, please visit [www.gleanster.com](http://www.gleanster.com).

## About Gleanster

Gleanster benchmarks best practices in technology-enabled business initiatives, delivering actionable insights that allow companies to make smart business decisions and match their needs with vendor solutions.

Gleanster research can be downloaded for free. All of it.

For more information, please visit [www.gleanster.com](http://www.gleanster.com).

### Headquarters

Gleanster, LLC  
4695 Chabot Drive  
Pleasanton, CA 94588

For customer support, please  
contact [support@gleanster.com](mailto:support@gleanster.com)

For sales information, please  
contact [sales@gleanster.com](mailto:sales@gleanster.com)

**Note: This document is intended for individual use. Electronic distribution via email or by posting on a personal website is in violation of the terms of use.**