Why Big Data is Critical for APM
Contents

Why Big Data is Critical for APM ................................................................. 5
  What is “big” about big data? ................................................................. 5
  What is the big data approach to APM? ................................................ 6
  Big data and artificial intelligence ...................................................... 6
  Big data and its human impact ............................................................. 7

Key Focus Areas for Big Data APM ......................................................... 9
  Classifying performance problems ..................................................... 9
  Diagnosing intermittent problems ...................................................... 9
  Visibility into highly dynamic environments .................................... 10
  Understanding the user journey ......................................................... 10
  Continuous performance improvement .............................................. 11

Myths About Big Data and APM .............................................................. 14
  Myth #1: Not scalable ....................................................................... 14
  Myth #2: Difficult to manage ............................................................ 14
  Myth #3: Prohibitive overhead .......................................................... 14
  Conclusion ....................................................................................... 15

About Riverbed SteelCentral ................................................................. 15
For application performance management (APM) to achieve its objectives in a world of composable applications and dynamic workloads, it needs granular data about both the workloads and their environment. This means that APM solutions need to enter the world of big data and advanced analytics.
Why Big Data is Critical for APM

It has largely been viewed that a big data approach to APM presents an insurmountable technical challenge. The primary obstacles cited are:

1. The performance burden on the application being measured, and,
2. The challenges that the volume, velocity, and variety of the data set present in terms of storage, processing, and data transfer.

Therefore, many vendors have focused their efforts on enhancing analytics, while relying on sampled and incomplete data sets for diagnostics.

Which is better? More algorithms or more data? This paper argues that when all relevant measurements are collected, you can significantly amplify the power of analytics. By combining the complete set of "big data" with analysis, you gain deeper and more accurate insight into application performance and take your application performance management practice beyond the level of merely monitoring and alerting.

What is "big" about big data?

Big data is so vast because all data, including raw and unfiltered data, is captured. It is an engineering feat that with big data technology IT can now collect, store, analyze, and access petabytes of data, but the most amazing aspect of it is that those petabytes of data represent the unfiltered source data – free from inaccuracies and blind spots due to sampling or averages. Any time data is sampled, filtered, or aggregated, the resulting record represents only a small portion of the real picture. By collecting, managing, and analyzing big data, you can better explore, understand, and draw insight from the data.

The primary benefit of big data is that it enables us to gain immediate insight without needing to first create a hypothesis, design sampling strategies, and run experiments to test the hypothesis. With big data, you can observe the entire universe of the problem and the resulting analysis is free from inherent sampling or selection bias.

---

### Characteristics of APM Big Data

<table>
<thead>
<tr>
<th>Volume</th>
<th>Petabytes of APM data generated daily must be transferred, processed, and stored</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity</td>
<td>Billions of transactions/day; application state is transient and must be measured at high frequency, ideally 1-second intervals</td>
</tr>
<tr>
<td>Variety</td>
<td>Every transaction is unique; data can be unstructured log data, performance metrics, transaction parameters, etc.</td>
</tr>
</tbody>
</table>

Table 1 APM big data presents collection, processing, and storage challenges due to the tremendous volume, velocity, and variety of the data.
What is the big data approach to APM?

A big data approach to APM collects all the relevant application workload and application environment metrics and then applies advanced analytics to surface insights. In contrast, sampling approaches collect transaction performance data less frequently and provide less granularity into the application stack and attempt to fill in the blanks with algorithms.

In general, APM data can be classified into two categories of measurements:

1. **Application workload data**
   - User sessions or actions
   - Application execution call stack (“tracing”)
   - Error messages
   - Application request and response parameters
   - Calls to external APIs
   - Application outputs, i.e. log data

By measuring the workload in the form of user actions or transactions, you can gain a greater understanding of how the application behaves and whether service levels are being met.

With workload data, the two key aspects to consider are the depth and completeness of the transaction record captured for each transaction. For example, capturing every user action alongside the full record of how the application responded across every part of the application stack gives a much more complete understanding of the workload than providing average responses for all users based on a sample set of data.

2. **Application environment data**
   - Infrastructure utilization
   - Virtualization and underlying system resources
   - Application processes
   - Container resources
   - Data sources
   - KPIs

With environment data, the two focus areas are the number of metrics and the frequency of data captured. For example, you will have better visibility into how the infrastructure is behaving if it is capturing data at second intervals versus polling every minute.

These key metrics provide a good indication of how the underlying infrastructure responds to the demands of the application whether it is within the data center or in the cloud(s).

There are nuances to how to manage and analyze the big data in each category, but in both cases, the end goal is to provide a complete and correct understanding of application performance and end-user experience.

**Big data and artificial intelligence**

There is a lot of excitement and focus around machine learning and artificial intelligence (AI) in the APM context. However, there is little mention of the foundation of AI: the data. Deep machine learning algorithms are inherently driven by data (also referred to as “training sets”). Bad data, such as incomplete or sampled data, leads to poor-quality AI recommendations.

“More data usually beats better algorithms.”

Dr. Anand Rajaraman
Assistant Professor, Stanford University

A couple of years ago, students in a Stanford data mining class participated in a competition to beat Netflix’s movie recommendation algorithm.¹ The winning team was able to come up with better recommendations based on access to more metadata (they appended data from IMBD to the data set they were given). In spite of running a very rudimentary algorithm, the team with more data performed far better than the teams that relied heavily on algorithms. Extending this to APM, a big data approach is expected to yield higher quality AI recommendations based on the richness of metadata that is available for pattern extraction.

¹ http://anand.typepad.com/datawocky/2008/03/more-data-usual.html
This is not meant to discount the role of artificial intelligence altogether. When AI and analytics are applied to a rich set of APM big data, you can quickly surface insights for troubleshooting. Two examples of machine learning algorithms for APM are:

1. **Pattern recognition**: Cluster recognition and correlation algorithms help you find groups of related transactions and performance metrics.

2. **Anomaly detection**: Anomaly detection proactively alerts you to anything that is exhibiting abnormal behavior before end-user SLAs are breached.

---

**No one sitting in a war room ever said, “We need less detail to find the root cause!”**

---

**Big data and its human impact**

Let’s examine a real-world performance incident in which a QA test is failing or a production environment is experiencing performance degradation on a critical application. The tiger team is quickly mobilized to analyze the problem and make a recommendation. The success of the triage effort dramatically improves depending on the quality and the completeness of the forensic data given that each of the multiple possible root causes will require further research. Members of the team will often rely on assumptions or educated best guesses in the absence of data. With complete and accurate forensic data that describes the problem in depth, you can remove ambiguity, rally the team, and ultimately drive to faster resolution.
Key Focus Areas for Big Data APM

Classifying performance problems
Applications are often plagued with multiple performance problems. A big data approach helps IT divide and conquer the long tail of problems more efficiently and identify potentially multiple causes contributing to poor performance of a single application. Often the triage team spends valuable time chasing the symptoms and looking for a silver bullet, one root cause that when fixed resolves all the performance problems.

In one example, what seems like a single problem is actually 11 different problems that impact the performance of a business-critical application. With a complete record of how the application is behaving, the applications team is able to identify which incidents are connected to each part of the problem and prioritize fixes accordingly. Also, because these 11 problems are in different parts of the application, much of the work could be done in parallel. In the absence of big data, different problems often get categorized as the same problem, which, in turn, leads the triage team down the wrong path.

Definitive analysis
Big data removes the would/could/should from the performance analysis. In the absence of precise data, performance analysis may hinge on experience, which may or may not be applicable. Let’s look at an example with sampling data. Application performance reaches a threshold and this triggers a more detailed snapshot of the offending transaction. We are able to tell where the transaction is delayed, but we are not able to tell why it is delayed because there is not enough information about the state and the activity in the rest of the application. This often occurs when there is a shared resource or shared code in an application environment. Often depleting that shared resource will impact applications disproportionally and the applications that deplete the resource may not be the ones that show symptoms first. For example, an infrequently used application that consumes all of the connection pool resources will fly under the radar, and the highly-used application that heavily relies on that pool will trigger first by exhibiting a large volume of stalled transactions.

In the investigation process, it is just as important to determine what is not the cause of a problem.

Often when doing analysis, the triage team will be tempted to use prior knowledge in the absence of forensic detail – “Last time we had a performance problem it was our logging code” – and it would frequently go down the wrong path. With big data, we can quickly say, “It’s not the logging code” with confidence because every data point is captured. Since there is no record the logging code is being used, the team can move on without wasting time or effort.

Diagnosing intermittent problems
Intermittent performance problems tend to be the most challenging to diagnose for several reasons:

- The conditions of the failure are often elusive
- Recurrence is unpredictable
- There are few opportunities to observe the problem
- The environment itself is changing throughout the duration of these long-running problems
A big data approach addresses all these challenges and enables IT to quickly diagnose challenging, intermittent problems. With big data, it is not necessary to understand the failure of conditions up front since the diagnostics data is continuously captured in full detail. For the same reason, there is always forensic data available to aid in root cause analysis regardless of when the problem occurs and how the application environment has changed. On the other hand, with sampling or triggered collections, intermittent problems may be virtually impossible to solve.

Visibility into highly dynamic environments

A big data approach is very effective in diagnosing problems in cloud, virtualized, or containerized environments. In these ephemeral application environments, the application infrastructure is constantly changing, and a triggered/sampled approach may miss the state changes, which occur as components are spun up and spun down.

Every transaction matters. One percent of transactions performing poorly does not mean that 99 percent of users are happy.

Typically, an application component like a microservice is spun up to process user requests on a newly provisioned cloud instance. After application demand subsides, the instance is removed. In a sampled approach, the state of the new node may never get recorded, or it may get occasionally recorded in application traces. This could easily mislead you to conclude that a remote call was made to a node that was not available instead of correctly determining that it was a call made to an existing node with a code-related performance problem. In this case, there may be some discussion as to whether the data is missing because it wasn’t recorded or because it wasn’t there in the first place. However, with a continuous measurement approach and complete transaction record of the stack, there is no ambiguity as to the state of the ephemeral nodes.

Using big data eliminates this guesswork and allows you to capture every state transition in order to accurately portray how a system behaves. This is also known as the first Law of Cybernetics, the law of Requisite Variety, or Ashby’s Law. It states, “If a system is to be stable, the number of states of its control mechanism must be greater than or equal to the number of states in the system being controlled.” What Ashby’s Law implies is that the degree of control of a system is proportional to the amount of information available. Simply put, this means you need an appropriate amount of information to control any system.\(^2\)

Understanding the user journey

Understanding the user population is invaluable in drawing insight into global performance trends but is sometimes insufficient in understanding the steps that could lead to significant performance problems. A single user action can lead to performance problems for the entire application and a big data approach guarantees that all forensic data is available to reconstruct the breadcrumb of the incident.

Consider this scenario in an e-commerce company where one user accessing from one type of web browser caused an entire business-critical application to crash. The company experienced sporadic JVM degradations caused by runaway threads on their web servers, which gradually consumed all the available resources on the machine. Application Support identified the offending application, but it was not clear what caused the issue. The team then analyzed how users were accessing the application and identified a single user session from a specific version of Safari from a mobile device that caused Apache to go into an infinite loop. Another scenario is when there is a single malicious user who can compromise data and lead to an outage. Understanding the individual actions that led up to the incident can help thwart any further damage.

\(^2\) Learn more at https://firstlaw.wordpress.com/2011/10/18/ashbys-law/
If you sample or disregard the performance of only one percent of transactions, it could have immediate bottom line impact. It is a bit counterintuitive, but one percent of transactions performing poorly does not mean that 99 percent of users are happy. How is this possible? A single user session is comprised of hundreds of transactions. For example, a user would browse, look at items, compare, checkout, etc. and each would be a separate transaction. It follows that if on average one percent of transactions perform poorly, statistically it could mean that every single user session will have a negative experience along the way.

**Continuous performance improvement**

Big data is more than just monitoring and diagnostics. It can be used to methodically reduce performance bloat and provide the basis for forensic exploration or code audit. The availability of deep performance data becomes the basis for continuous performance improvement.

**The result is that over time a well-performing application starts degrading**

Applications are continually changing with new feature releases and tend to accumulate technical and performance debt.

With sufficient performance data and big data analytics, DevOps teams and developers can better understand where time is being spent.

How does it work? The full call stack for each transaction contains details on how much time is spent at each step of the application code. With big data analytics, IT can aggregate all this data to answer definitively, which methods/SQL/external are costing the most in terms of performance. Take for example a critical application that needs to be put on a “diet.” The analytics can identify the top calls that cost the most across the entire application using production performance data. The DevOps team may suspect they know where the most significant issues are – however, the results may surprise them (i.e. when one very common logging facility appears on the top of the list). On aggregate, more data can often overturn assumptions about the most expensive application calls.

In an enterprise setting, applications are never designed, built, or operated in complete isolation. In some cases, different applications may share systems, networks, or infrastructure. In other cases, applications may share common libraries, data, or APIs. Sharing of components or resources accelerates development cycles and lowers costs but also leads to performance problems that impact multiple applications.
Visibility into the entire application environment

Big data helps you see the big picture. With it, you can uncover performance problems and patterns that impact across the entire application environment not just within a single application component. Once a problem is discovered in a single application, analytics applied to this big data can help identify other applications that have the same problem or are at risk.

Take for example a SQL database call that is causing performance degradations in a particular application. Using big data analytics, the Applications Support team can leverage the relationships discovered with big data analytics and uncover other applications that are using the same SQL statement. If this is a common weakness, it can be quickly addressed and the problem eliminated across the board. To illustrate this point further, take the example of a company preparing for a major launch but cannot not pass their performance test in QA. The load test is conducted using synthetic user sessions that step through multiple transactions traversing multiple applications. Based on the test results, there are multiple candidates to optimize, but after analyzing the entire data set, a single external web service call stands out as the biggest bottleneck across all applications. Optimizing the shared dependency has the most significant impact on the overall system performance and avoids cycles spent optimizing each application individually.

Image 1: Immediately see, which pieces of code contribute to the slowest user transactions with a performance graph that simplifies how you visualize complex application dependencies.
Myths About Big Data and APM

There are several well-accepted myths in the industry with regards to whether a big data approach to APM is even feasible. Scalability, manageability, and agent overhead are frequently cited as reasons to avoid a big data approach to APM. However, many of the technical concerns around big data have already been addressed for other markets and these paradigms can be successfully applied to APM.

Myth #1: Not scalable

Capturing detailed transaction records for every user action and backend activity does generate a lot of data, but APM is no different than other use cases for big data in this respect. The same techniques and architecture apply to APM, for example, leveraging highly optimized, non-relational data stores. Big data APM typically uses proven streaming and complex event processing architectures. Just as in Hadoop clustering, the APM big data can also federate data processing and analytics across the APM components. With a three-tier processing approach, the in-app collectors do minimal work and pass the raw data to a local processor, which then sends the data to the central engine after performing compression and some base analytics. This approach scales far better than a two-tiered approach with in-app collectors that can tax the application and/or the backend system.

Myth #2: Difficult to manage

Continuously capturing all transactions in detail results in a monitoring system that is far more resilient and easier to manage because there is no reliance on complex rules, inspection, or a trigger engine that needs to be configured and maintained. There is undoubtedly more data persisted with this approach but managing storage is a technical problem that has been largely solved at this point. It is also important to recognize that storage is a much less expensive resource than adding compute resources.

Myth #3: Prohibitive overhead

Overhead is a key concern for any technology that is sitting in line with the application and APM agents are no different. Continuously capturing transactions to enable big data APM necessitates different techniques for how transactions are observed, recorded, and persisted to ensure that the application’s performance does not suffer from instrumentation overhead. To minimize overhead, agents can dynamically discover every component in the application stack, record every critical method and application call in a transaction, and then compress and stream transaction data. Using a three-tier architecture for big data APM eliminates the need to add compute or processing power to the application stack being monitored.
Conclusion

Unlike data sampling or use of triggers, a big data approach to APM ensures that all the data is available when and as needed. APM big data can then be used effectively to resolve performance problems faster and optimize performance, even in dynamic application environments that rely on microservices, virtualization, and cloud services.

The techniques described in this paper have been successfully applied and tested in real-world scenarios using the Riverbed SteelCentral Application Performance Management (APM) suite. SteelCentral leverages unique, patented technology for big data APM to help customers reduce mean time to repair and improve overall performance.

About Riverbed SteelCentral

Riverbed® SteelCentral™ Application Performance Management blends end-user experience, application, and network monitoring to provide end-to-end visibility into the users’ digital experience and ensures the reliability of business-critical applications whether they are running on mobile, virtual, or physical devices. Riverbed’s solutions can be deployed on-premises or in the cloud, enabling customers to get up and running fast, with no major capital investment, hardware provisioning, or server deployment.

To learn more about SteelCentral AppInternals, visit www.riverbed.com/appinternals and sign up for a free trial.
About Riverbed
Riverbed®, The Digital Performance Company™, enables organizations to maximize digital performance across every aspect of their business, allowing customers to rethink possible. Riverbed’s unified and integrated Digital Performance Platform™ brings together a powerful combination of Digital Experience, Cloud Networking and Cloud Edge solutions that provides a modern IT architecture for the digital enterprise, delivering new levels of operational agility and dramatically accelerating business performance and outcomes. At more than $1 billion in annual revenue, Riverbed’s 30,000+ customers include 98% of the Fortune 100 and 100% of the Forbes Global 100. Learn more at riverbed.com.