



# Research Report

## How New Business Transaction Management Tools Are Making It Possible to Address Transaction Performance Tuning and Capacity Planning Issues in the Cloud

### *Executive Summary*

What happens when a transaction is sent into a cloud for processing? Which physical and virtual resources does it use (how do you do capacity planning in a cloud if you don't know which resources a given transaction is using)? What dependencies does the transaction have? If the transaction is performing poorly, how can the fault be isolated? If the transaction misbehaves intermittently, how can that fault be isolated? And, how do you tune transactions in the cloud to improve performance?

*Tracking application and transaction faults in a distributed systems environment or a cloud can be bedeviling. It involves following applications and transactions as they hop from server to server making database calls and performing reads and writes along the way. It also involves tracking business process flows to ensure that transactions execute efficiently. For enterprises that are using the wrong tools, tracking transactions and monitoring/managing business process flows in a cloud architecture can be extremely time-consuming and fraught with organizational finger-pointing.*

### *The Evolving Business Transaction Management (BTM) Discipline*

The practice of tracking transactional behavior across information systems is called business transaction management (BTM) — a segment of the application performance management (APM) marketplace. Many vendors offer BTM software solutions that can troubleshoot, control, and manage for transactions within cloud environments. But there are very significant differences in the approaches that these vendors use (some vendors can only measure activities in silos using a network sniffer/time-based correlation approach to find problems; others use a more automated, deterministic, transaction monitoring/-management approach that can span across server tiers while providing a topological overlay of transaction behavior that represents how a transaction is behaving in real time).

In this *Research Report*, *Clabby Analytics* examines the way that transactions are handled in distributed computing/cloud environments. We closely examine the network sniffer/time-based correlation and real-time, deterministic cross-tier approaches — and we conclude that the network sniffer/time-based correlation approach is “architecturally challenged”. Accordingly, ***we recommend that enterprises that are looking employ cloud architectures to handle high-volume transaction processing use topographically-driven, cross-tier, deterministic, real-time application/transaction BTM management tools.***

### *Background*

Until recently, *Clabby Analytics* would have argued that cloud computing architecture has two major shortcomings: transaction handling and security. Our argument would have

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

been based upon our experiences in grid computing, an architecture that has a lot of similarities to cloud computing. Back in 2002-2005, *Clabby Analytics* closely followed the grids, and we found grids to be well suited for handling behind-the-firewall scientific/technical applications — and for addressing business problems that could be parallelized across many computers where the conclusion of each parallel stream could be reassembled after processing to produce a final result. But we also found several grid shortcomings related to the processing of certain types of workloads, including:

- Complex batch applications involving a lot of data transfer;
- Heavily transaction-oriented applications (where transactions needed to be tracked and monitored across grid servers — and potentially rolled-back in case of failure);
- Applications where external databases needed to be accessed regularly; and,
- Application environments where strong authentication/authorization is required.

One of the biggest problems with early grid designs was that grid resource management software could not effectively track application behaviors as applications traversed through various grid-connected systems (for instance, processing a multi-step transaction might involve the use of computing resources in sales, accounts receivable, manufacturing, and shipping departments). If an application or transaction failed as it progressed through a grid, much manual work/guesswork was required track-down and isolate the source of the problem, and then fix the problem.

Because tracking down application/transaction failures was so complex, grid users gravitated toward predictable, parallel workloads such as those found in scientific environments. In these environments, if a segment of an application failed, it could be easily be restarted and reassigned to other computing resources. (Examples: the SETI@Home project, or cancer research project Folding@Home).

*Contrast these parallel workload projects with transactional workloads. Transactions are contiguous workflows that may involve ten or twenty hops (or more) across various servers in order to reach completion. Further, failed transactions require roll-back — a sometimes “tricky” task. Clearly, managing multi-hop applications across a cloud environment is far more complex than simply restarting failed parallel processes. This degree of complexity is part of the reason why the move to grid computing stalled when it attempted to move into the commercial, transaction processing market place. And, before the arrival of BTM tools, clouds suffered from the same limitations.*

### ***Market/Competitive Overview: The BTM Marketplace***

As stated earlier, BTM is an evolving market — an offshoot of the APM market. And, as is the case with many evolving markets, a definitive definition of this discipline is hard to come by, primarily because vendors in this market tend to bend the definition to fit their tools and their approach to transaction management. A discussion of the Gartner, EMA, and IDC definitions of BTM — as well as several vendor definitions — can be found here: <http://businesstransactionmanagement.blogspot.com/2008/10/definition-of-business-transaction.html>. Further, a comprehensive discussion of BTM can also be found here: <http://dougmcclure.net/blog/business-transaction-management/>.

### ***BTM Vendor Approaches***

There are dozens of vendors that offer application management products. All of these vendors also offer products designed to track and monitor transactional behavior. But there

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

are huge differences in how some vendors handle transaction tracking and monitoring versus others.

In general, BTM approaches fall into two camps:

1. A *siloe*d application performance management approach that uses a network sniffer/time-correlation approach (also called a “network tap” approach) to analyze transaction behavior. (Examples of products in this category include CA’s NetQoS, HP RUM, and CA Wily CEM). These products are useful in troubleshooting products within a single server — but are cumbersome when trying to trace transaction behavior across multiple server tiers; and,
2. A *tiered* application performance management approach that uses topology-driven, deterministic tracking/monitoring approach.

### *An Overview of BTM Vendors*

A partial list of vendors that offer BTM products includes:

1. Amberpoint — Amberpoint Management System;
2. BMC — Transaction Management;
3. CA — Wily Introscope, NetQoS, and Customer Experience Manager (CEM);
4. Compuware — Agentless and Agent-based Vantage for Transaction Profiling;
5. Dynatrace — dynatrace;
6. Hewlett-Packard (HP) — TransactionVision;
7. IBM — IBM Tivoli Composite Application Manager;
8. Jinspired — Jxinsight;
9. Precise — Precise Transaction Performance Management;
10. Nastel Technologies;
11. MQSoftware — Q!Pasa,
12. Opnet — Panorama;
13. OpTier — Corefirst, User Experience Manager; and,
14. Oracle — Quickvision.

Of these vendors, Dynatrace, Jinspired, and Quest appear to fit into the sniffer/time correlation category. The remainder fit (to various degrees) into the deterministic category.

*As examples, HP’s TransactionVision, MQSoftware’s Q!Pasa, CA’s Wily Introscope, OpTier’s Core First combined with its User Experience Manager, and a variety of IBM’s Tivoli products do instance level transaction monitoring using a deterministic approach. These vendors offer products that can thread together and topologically represent an entire transaction as those transactions make their way through multiple systems in a cloud. Using this approach, cloud managers and administrators can view an accurate representation of transactional dependencies, transaction resource usage, transaction service levels, and other elements that pertain to transaction management. And with this data, these managers and administrators can isolate problems in the application, database, or middleware layers — and bring in the right human resources (from application/database/infrastructure organizations) to fix those problems.*

### *A Closer Look at the Siloe*d, Network Sniffer/Time Correlation Approach

When it comes to transaction processing in clouds, one of the biggest challenges is tracking a transaction’s behavior as it wends its way across several servers in a cloud environment. To trace a transaction’s behavior in a cloud, several vendors offer “network sniffer” and

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

“correlation” tools that monitor events in a given environment, aggregate results, and then correlate those events with other events elsewhere in the environment in order to determine the cause of a transaction failure or poor application/transaction response time.

*The big problem with using this sniffer approach is that a sniffer cannot track a specific transaction as it traverses a network. It can collect snippets of a transaction flow — and those snippets can be used to help isolate the location of a server or a process on a server that is experiencing a resource consumption spike or service elongation — but a sniffer cannot weave these snippets into a cohesive view of how a transaction is behaving as it multi-hops through a morass of servers. By not being able to monitor a specific transaction flow as it makes its way through a distributed computing/ cloud environment, discrete transaction problems that are low volume or that hang anywhere in a transaction life-cycle are missed. Extra effort (and sometimes team efforts) is required to help locate the source of these types of problems.*

As an example of this approach, consider how a network sniffer is used to examine response time violations. Network sniffers are used to diagnose network-centric problems. During a diagnostic exercise, sniffers run continuously, collecting an enormous amount of information about the impact that a given application is having on the network. This data can then be analyzed to help determine where an application becomes slowed — helping IT managers determine which server is responsible for a particular delay.

Once analysts determine the location of a problem, they need additional insight into a resource’s behavior in order to determine the cause of that problem. To do this, IT managers who use sniffer tools then use a “time correlation” approach to look for corresponding events that occurred around the same time as a transactional failure that may have contributed to the failure. This approach uses aggregations of service level violations, and aggregations of time-correlated back-end “bad behaviors” to help determine why a problem is occurring.

*The sniffer/correlation method has been used for years — but, this approach can prove to be very time consuming and results can be ambiguous. The accuracy of a correlation can always be questioned (checking the time stamp of a given failure doesn’t necessarily mean that the true failure has been correctly isolated and identified). From Clabby Analytics’ perspective, the NETWORK SNIFFER/ CORRELATION APPROACH IS WOEFULLY INADEQUATE WHEN IT COMES TO MANAGING TRANSACTION FLOW IN CLOUD ENVIRONMENTS. Network sniffer-based monitors do not track individual transactions across servers. And using the sampling/correlation approach for troubleshooting (an approach that essentially looks at time stamps when a transactional failure occurs — and then tries to find a corresponding failure that occurred at the same time somewhere else in the cloud — and then correlates the two events) amounts to little more than educated guesswork.*

### ***The Cross-Tier Deterministic Monitoring/Management Approach***

A growing list of vendors, however, (including large vendors such as CA, IBM, and Hewlett-Packard — and smaller vendors such as CorrelSense and OpTier) are using a completely different approach to troubleshoot, monitor, manage, and tune in cloud environments.

*This alternative approach uses topology mapping and provides direct, real-time transaction monitoring and measurement as opposed to sniffing/correlation to unambiguously trace the causes of transaction failures or performance problems. To be more precise, this deterministic monitoring/management approach uses a persistent modeler to determine the cause of transaction failures. This modeler can:*

- *Paint a graphical, topological representation of a transaction as it crosses server environments;*

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

- *Illustrate all of the processes (and dependencies) involved as a transaction makes its way through a processing tiers; and,*
- *Clearly and unambiguously identify the cause of a failure (for instance, a failed computer component or an application failure if a transaction takes an unexpected path).*

For further insight into how this approach works, consider Hewlett-Packard's (HP's) TransactionVision. The TransactionVision patent states that: "the 'correlation' described in this patent is NOT a statistical correlation, but a deterministic correlation based on transactional metadata captured at the agent". In other words, a correlation is taking place — but not the same kind of correlation that takes place using the sniffer/time-correlation method. In this case, the correlation is the result of agent activity that captures application behavior and submits it to a metadatabase where it can be described and analyzed.

### *Contrasting the Sniffer/Correlation and Deterministic Approaches: Some Real World Situations*

In addition to being simpler to use while providing insight across multiple tiers, cross-tier cross-tier BTM products can trace problems that siloed tools struggle to find. For instance:

- If a server tier is overly busy (due to increased CPU and I/O usage), page faults may result. Both the sniffer/correlation and deterministic methods can isolate the cause of a failure in a busy server tier environment. But, if a transaction stalls or fails in a tier and does not progress to other tiers, then the deterministic approach has a clear advantage over the sniffer/correlation approach. Deterministic environments track individual units of work — and can explicitly enumerate cases where the unit of work count is less. The sniffer/correlation approach assigns a baseline value and "guesses" at the work count results.

*Normally a transaction transits 5 different tiers over, say, 30 units of work (multiple calls per server). If a failure occurs while a transaction is executing, and it completes only 10 of the 30 units of work, the deterministic approach counts/monitors discrete units of work — and can determine that only part of the job has been completed.*

- If an application artificially serializes (gets out of synchronization) because of a logic/serialization problem or a permissible thread value set wrong, the sniffer/correlation approach would have trouble identifying the source of such a failure, whereas the deterministic approach would be tracking it from the onset of the application to the point where the application failed.
- If a transaction takes an unexpected path because of a logic failure (example: including a test database in a production code promotion), a network sniffer would not have a clue unless the underlying transaction was also a major consumer of resources.
- BTM tools also have the ability to look at asynchronous and long-lived transactions — a capability that APM tools generally do not have. Most APM vendors grew up looking at web traffic and from that point of view, a transactions rarely lasts more than a few seconds. But, with BTM, a transaction can span the entire enterprise and be more complex with a much longer lifespan. BTM tools can easily accommodate the concept of long-lived transactions.

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

In addition to helping to find transactional faults, BTM tools can be used to prove that applications are behaving properly. As an analogy, consider an express mail tracking system. Usually tracking systems are used to find out the status of a delivery, or to show where a delivery problem exists. But, conversely, the same tracking system can show a positive result such as proof that a package has been delivered to the right place on time. HP's TransactionVision, BMC through MQSoftware, OpTier, and Nastel are BTM providers that provide such "proof of innocence" today.

### *The Organization Impact of the Deterministic Tools*

Within the enterprise, there are four "stakeholder" organizations that need to understand the differences between these approaches. They are:

1. *Performance analysts* (who analyze systems, storage, network, application, and database behaviors);
2. *Capacity planners* (who need to understand the resource utilization characteristics of applications and transactions in order to ensure that there is enough capacity to meet current and future transaction thresholds);
3. *Application developers* (who need to understand where applications get "hung-up" in order to adjust their programs to perform better); and,
4. *Business planners* (the business facing community that needs to understand if applications and transactions are meeting expected service levels [note: this group needs to be able to document service level failures in order to negotiate penalties should a service provider not meet its service level]).

***One of the biggest problems with the sniffer/correlation approach is that the root cause of application/transaction processing problems can be ambiguous. And ambiguity can cause organizational problems — leading to finger-pointed and lost productivity amongst these organizations.***

### *Overcoming Ambiguity*

Imagine this situation: a transaction takes hours to complete as it traverses a cloud. The cause of this problem can be systems/storage/network-related, it can be database related, it can be related to the application design — or it can be related to prioritization or capacity problems. In other words, the blame for a transactional failure/delay can rest with any of a number of individuals from within a variety of organizations — including the application development organization and IT administrators and managers.

***Now consider this: during the course of our research we found a major financial institution that claims that the sniffer/correlation approach is capable of finding a problematic server (tier) when tracing transaction s only about 60% of the time. But the deterministic, integrated, topology mapping approach is able to find failures in servers, infrastructure, and applications 95% or more of the time.***

In situations where ambiguous results make it difficult to determine fault, siloed organizations (database, application development, OS, and network groups) often assume defensive positions. Organizational representatives are assembled to determine the cause of a problem — but the goal of individual team members is often geared to seeking ways to deflect the blame for transaction/application processing problems from their respective organizations. This defensive behavior is suboptimal because the emphasis shifts from all interested parties looking to solve a problem to parties looking to assign blame (and thus not focusing on actual problem determination and problem solving).

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

*Deterministic tools provide an unambiguous view of the source and cause of a given problem. Sniffer tools can help find the source of a problem and isolate the component responsible, but sniffers can not necessarily determine the root cause within a given component. By providing a deterministic, unambiguous view of a problem, the owner of that particular component/silo (DB, app, OS, network) can't wiggle out of that problem (and everyone else in the room knows it!). Accordingly, the identified stakeholder is strongly incented to really put some effort into problem resolution (rather than defending the honor of his or her respective organization).*

### *The Downside of Deterministic Tools*

Deterministic tools reduce ambiguity — saving IT managers and administrators time while leading to better organizational performance. But IT executives need to know that these tools are constantly operational as they monitor application/transaction behavior in real time. And, accordingly, there is processing overhead associated with the use of these tools. One financial institution characterized the impact of one of the deterministic tool maker's (OpTier's) CPU overhead as follows:

*“Our own experience, based on internal lab tests is the OpTier/CoreFirst offering causes a 2% increase in the in-path consumption of CPU and 3% degradation in throughput (at saturation). The in-path component of OpTier/CoreFirst, called a Tier Extension, has been made quite efficient. Its job is to be efficient, fast, stupid, and reliable, and appears to have succeeded in these objectives. All the other complexity/overhead is in the assembly and analysis of the transaction path, done in the CoreFirst Management Server, and outside the code path...”*

### *Choosing the Right Vendor to Monitor Cloud Transactions: Buying Criteria*

When choosing a BTM vendor, IT buyers need to evaluate the vendor's approach to transaction tracking/monitoring; the level of integration of that vendor's BTM product suite; and the depth of that product suite. Each of these is examined more closely below.

#### *The Approach*

The primary difference between the siloed approach to transaction management and the tiered approach is in the level of effort it takes to determine the cause of a problem. The siloed approach provides a view of what is happening on a particular server — hence, considerable effort needs to be expended to track transactional behavior across multiple tiers. The tiered approach provides a view across multiple tiers, simplifying transaction tracking and problem determination.

#### *Level of Integration*

Some vendors offer all of the right tools to track and monitor transaction behavior, to perform capacity planning analysis, and to tune transaction performance. But several vendors in the BTM camp have much work to do to integrate the flow between these tools. In some cases, achieving a management result involves launching several disparate applications and then manually sorting out data obtained by each tool to reach a given goal (such as performance optimization). Clearly, this approach adds complexity and is prone to errors — hence, integration is an important element in evaluating BTM offerings.

#### *Product Depth*

Some vendors have deeper BTM product suites than others. For instance, some track Web service flows through a given distributed computing environment, but provide limited visibility into other activities taking place across tiered servers. Others are good at

## How BTM Helps Address Transaction Performance and Capacity Planning in the Cloud

analyzing message queuing behavior — but little else. Still others provide good performance management and capacity planning tools that are deep and well integrated in terms of function. IT buyers need to determine which features and functions are most important to their respective information systems environments — and then chose BTM vendors that can best suit their particular computing needs.

### *Summary Observations:*

Cloud computing can help enterprises make better use of its investments in information technologies, scale capacity more easily, and lower operational costs. And, cloud computing can also enable enterprises to take advantage of new, more flexible pricing models. But, along with these benefits come new challenges such as tracking where transactions are running “somewhere in the cloud”, and capacity planning and performance tuning given difficulties in tracking how much of a given computing resource an application or database is consuming in a cloud.

New, deterministic transaction monitoring/management BTM tools are now available to help IT managers more efficiently and effectively manage existing distributed computing environments and evolving cloud environments. Additionally, these tools also make it possible to prove to stakeholders in application development, infrastructure, and information systems organizations that a given problem is unambiguously *theirs* to fix.

When evaluating BTM tools, ensure that the tools that your enterprise evaluates are capable of monitoring/managing transactions *across multiple tiers* (this will simplify troubleshooting). This point is critical because troubleshooting transactional problems very often involves traversing many server tiers (and the BTM tools offered by several vendors do not have this cross-tier capability). Also, make sure the tool that your organization chooses is capable of capturing resource consumption at each tier a transaction traverses (this is important for cloud performance management and capacity planning).

Finally, it is important to recognize that there is a close tie between APM, BTM and BPM (business process management). APM has been generally siloed within a single tier of the transaction flow (end-user, J2EE/.Net, messaging, CICS). BTM ties together the silos of APM to make a full enterprise view of the transaction. And this union enables IT managers to better understand automated process flows. Accordingly, deterministic, cross-tier BTM tools are becoming increasingly important to enterprises that are looking for ways to build cloud environments that can automatically and efficiently flow their business processes.

---

**Clabby Analytics**  
**<http://www.clabbyanalytics.com>**  
**Telephone: 001 (207) 846-6662**

*Clabby Analytics is an independent technology research firm. We focus our research on systems, storage, networks, infrastructure, management, and cloud computing. For more reports, visit: [www.ClabbyAnalytics.com](http://www.ClabbyAnalytics.com).*