Rethinking MTTR

Jim Metzler
Ashton, Metzler & Associates
jim@ashtonmetzler.com

Introduction

Somewhat tongue in cheek, I was thinking of titling this IT Impact brief "So many acronyms, so little time". I say that because based on whom you talk to, MTTR could refer to the:

- Mean time to repair a problem
- Mean time to restore a service
- Mean time to respond to a trouble

Throughout this brief, MTTR will refer to the mean or average time that it takes the IT organization to repair a problem.

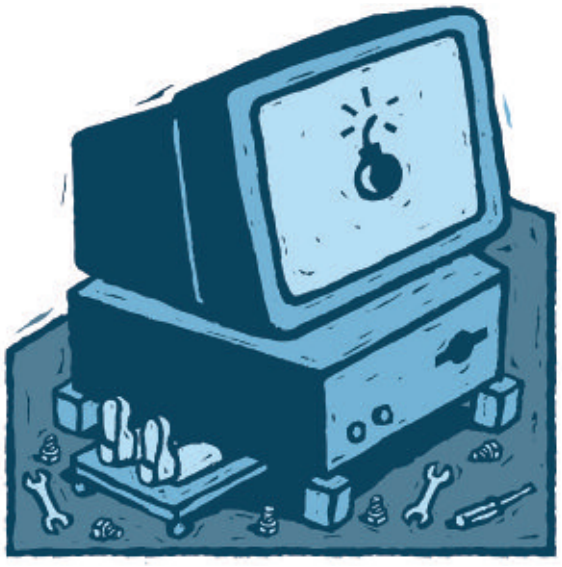
- Problem Identification
- Problem Diagnosis
- Solution Selection and Repair

The goal of this brief is to describe how those steps apply to a traditional network management task, such as fault management, and to contrast that with how those steps apply to managing application performance.

In March we conducted a survey on a variety of topics, including troubleshooting. Throughout this IT Impact Brief the almost 300 members of the NetScout community who responded to that survey will be referred to as The Survey Respondents. To gain additional depth into the topics that will be covered in this brief, four members of the NetScout community were interviewed. Table 1 contains some background information on the interviewees.

Industry	Title
Education	Director of Cyber Infrastructure
Telecommunications	Manager
Financial	Network Engineer
Manufacturing	Senior MIS Specialist

Table 1
The Interviewees



Throughout this IT Impact Brief, the interviewees will be referred to as The Education Director, The Telecommunications Manager, The Financial Engineer and The Manufacturing Specialist.

MTTR and Managing by Objective

The interviewees represented a wide range of approaches to MTTR. The Manufacturing Specialist stated that his organization does not officially measure MTTR, but that they do estimate MTTR. The Telecommunications Manager stated that his organization pays a lot of attention to MTTR, but that it applies only to the availability of the infrastructure and the applications. The Education Director stated that his organization does measure MTTR, but that they do it only for fault management and not for application performance. He also stated that currently the MTTR is around 3 or 4 hours, but that his management is getting more demanding and wants him to reduce the MTTR.

The situation reported by The Financial Engineer was more sophisticated. Not only does his organization measure MTTR for both availability and application performance, they compute separate MTTR metrics for different priorities of troubles. For example, the highest priority is a trouble that causes a large number of users to not be able to access the information or applications that they need to do their jobs. The next highest priority is a trouble that results in a large number of users being able to access the information and applications they need, but in a degraded fashion.

The Survey Respondents were asked if they or someone in their organization has an MBO (Management by Objective) related to mean time to repair. Their responses are shown in Figure 1.

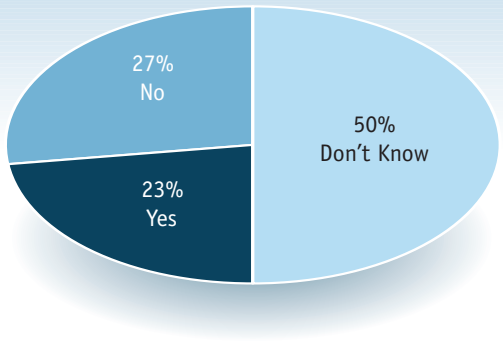


Figure 1
Having an MBO for MTTR

One observation that can be drawn from Figure 1 is that roughly the same number of companies has an MBO for MTTR as don't have one. Perhaps more interesting is the fact that half of The Survey Respondents answered 'don't know'. There are a variety of ways to interpret that response. For example, it could be that someone in the organization does have an MBO for MTTR but does not take it seriously. Alternatively, it could be that someone in the organization has an MBO for MTTR, takes it seriously, but does not advertise the fact that they have the MBO. An interesting trend uncovered was that the percentage of people who have an MBO related to MTTR is on the rise, growing from 6.3% of the responders in a January 2006 survey to 23.3% in this March 2007 survey.

Fault and Application Management

According to Wikipedia, fault management is the set of functions that detect, isolate, and correct malfunctions in a telecommunications network, compensate for environmental changes, and include maintaining and examining error logs, accepting and acting on error detection notifications, tracing and identifying faults, carrying out sequences of diagnostics tests, correcting faults, reporting error conditions, and localizing and tracing faults by examining and manipulating database information.

As will be discussed below, the troubleshooting process associated with managing application performance is significantly different than the troubleshooting process associated with fault management. That does not mean that MTTR is no longer important, but that we need to re-think how we measure and use the information. It also means that we need to take proactive measures to shorten the time it takes to perform each of the three steps that IT organizations go through in order to troubleshoot a trouble.

Recent IT Impact Briefs have detailed the issues involved in managing application performance, including the January 2007 brief, The Road to Successful Application Delivery and the February 2007 brief Management and Application Delivery.

The Manufacturing Specialist commented that within his organization managing application performance is a shared responsibility. He noted, however, that because of the success they have had with the management tools that they have deployed, other organizations call them seeking help with resolving a problem. The Manufacturing Specialist also pointed out that while adding the capability of managing application performance is important, "you can not lose track of existing issues." He added that fault management is still important and that they "are still tracking IOS bugs."

The Financial Engineer stated that his organization handles application performance issues pretty much the same way that they handle issues that impact availability. He added that his organization has received significant additional training in part to be able to better manage application performance. To put this in context, he pointed out that his company takes training very seriously and that if employees do not achieve their yearly goals for training that "their manager gets dinged for it on their review."

The Telecommunications Manager stated that they have begun to measure application degradation. He added that application degradation is taken more seriously inside of his company if he can quantify how much revenue was lost as a result of the degradation. The Telecommunications Manager also stated that his organization has begun to become ISO9001 certified. He expects that their level of certification will increase as the demands for better application performance increases.

The Manufacturing Specialist reinforced the importance of effective processes. He stated that in order to get better at managing the performance of applications, that his organization has implemented ITIL-based processes. He stated that these processes are "a huge part of our success" and the processes force you to "understand what you are doing and why you are doing it."

Problem Identification

Like every component of network management, fault management can either be done proactively or reactively. In a proactive approach, the network management organization attempts to identify and resolve problems before they impact end users. In a reactive approach, network management organizations respond to the fault once end users have been impacted.

One of the factors that make identifying a fault relatively easy is that a fault often leads to an outage, which is readily noticeable. Another factor is that it is relatively easy to set alarms to indicate that some component of the IT infrastructure has failed. In contrast, identifying that an application has degraded is much more difficult. For example, most IT organizations do not have objectives for the performance of even their key, business-critical applications. In addition, few IT organizations monitor the end-to-end performance of their applications. As a result, the issue of whether or not an application has degraded is often highly subjective.

To better understand how IT organizations identify when one of their applications is degrading, we asked The Survey Respondents to indicate how they determine when a performance problem requires immediate intervention. Their responses are shown in Table 2.

Trigger	% Respondents
When users complain	75.4%
When I receive an alarm	61.9%
When the network is more than 75% of capacity	36.9%
When the network is 50 – 74% of capacity	16.5%

Table 2
Trigger that Drives Immediate Intervention

Source	% Respondents
Application issue	34.3%
Server issue	23.3%
User error	22.0%
Network issue	18.2%
Other	14.0%

Table 3
Source of Application Degradation

The fact that the most common response was 'When users complain' underscores the difficulty associated with identifying an application performance issue.

The Financial Engineer stated that when a user calls in and complains about the performance of an application a trouble ticket is opened. He also stated that, "The [MTTR] clock starts ticking when the ticket is opened and keeps ticking until the problem is resolved." In his organizations there are a couple of meanings of the phrase "the problem is resolved." One of these meanings is that the user is no longer impacted. Another meaning is that the source of the problem has been determined to be an issue with the application. In these cases, the trouble ticket is closed and they open what they refer to as a bug ticket.

The Financial Engineer added that in some cases, 'The MTTR can get pretty large." He added that roughly 60% of application performance issues take more than a day to resolve. In those cases in which the MTTR is getting large, his organization forms a group that they refer to as a Critical Action Team (CAT). The CAT is comprised of technical leads from multiple disciplines who come together to resolve the difficult technical problems.

Problem Diagnosis

In the case of fault management, the focus of diagnosis is to determine which component of the infrastructure is not working. Part of the difficulty of diagnosing a fault is that a single fault can cause a firestorm of alarms. I don't want to understate the difficulty of filtering out extraneous alarms to find the defective component. However, it is easier to identify the component of the infrastructure that is not functioning than it is to identify the factor that is causing an application to perform badly. One of the reasons that it is so difficult to diagnose the cause of application degradation is that each and every component of IT could cause the application to perform badly. This includes the network, the servers, the database and the application itself. This means that unlike fault management that tends to focus on one technology and on one organization, diagnosing the cause of application degradation crosses multiple technology and organizational boundaries. In general, most IT organizations do not have a track record of efficiently solving problems that cross multiple technology and organizational boundaries.

In order to validate the statements made in the preceding paragraph, we asked The Survey Respondents to indicate the source of their performance problems. Their responses are shown in Table 3.

The data in Table 3 certainly supports the statement that each and every component of IT could cause an application to perform badly. The data is also interesting given that in most organizations the network is assumed to be the source of application degradation. The data in Table 3 indicates that the network is one of the factors that are the least likely to cause application degradation.

The Manufacturing Specialist stated that before they deployed probes they spent a lot of their time defending the network. He noted that now that they have probes deployed, they can quickly identify if there is a network issue.

The Financial Engineer stated that in his last company, 90% of issues were originally identified as being network issues even though the reality was that only 10% of issues actually were network related. He pointed out that incorrectly assuming that the majority of issues are network related has the affect of increasing the amount of time it takes to accurately diagnose the problem. He recommended that when a problem is called into the help desk that the person calling in should be encouraged to describe the symptoms of the problem in detail, and not just what they think the source of the problem is.

To better understand the difficulty associated with diagnosing the cause of application degradation we asked The Survey Respondents to describe the most difficult application degradation problem that they ever had to diagnose. I choose the following three responses to that question as they closely reflected the other responses.

Response #1

One of the sites was complaining the network is slow. The problem was intermittent and did not cause an alert from any of our automated monitoring tools. The WAN link was not showing high latency or high utilization on any of the weekly trending reports. Complaints were pretty much ongoing for a couple of months but no problem was found by the NOC. Once our tier-3 team was assigned to find the root cause we sent personnel to the site and had one person back at the office. By watching traffic patterns all day plus having hands on site we determined that there were multiple problems - configuration errors on the PCs as well as user errors. We were able to clean up their environment and fix the "network slow" complaint.

Response #2

A multi-tier application that transfers large files was consolidated into a local data center. The site that previously had the servers hosted locally was now connected via Gigabit DWDM connection to a data center across town. Performance for the application was significantly degraded, even though bandwidth utilization was acceptably low on the DWDM link. Detailed analysis of the traffic via probe captures and applicationlayer trending revealed that much of the traffic was NFS (Network File System) related mounts and file transfers. NFS is extremely sensitive to latency, and even the relatively small addition of 6-8 ms of latency from the change of a switched LAN environment to a Gigabit connected distributed environment was enough to make performance unacceptable. Some of the storage tier was relocated back to the local site to resolve the problem, as optimizing the legacy application was not an option.

Response #3

The most stubborn application performance issue we have encountered has been the delivery of applications over the WAN via Citrix. We have approximately 170 remote sites connected via 512k MPLS circuits to a data center. Most of the remotes use a database application that was originally deployed via a fat client and which required a server at each remote site. To cut costs and management overhead, the databases were centralized and the applications delivered via Citrix. However, performance dropped sharply resulting in complaints from the user community and the lack of bandwidth was assumed to be the culprit. Using network management tools, I was able to show that the bandwidth consumption was well within acceptable levels (averaging 30% with spikes to 60%), packet loss was minimal, MRTTs were acceptable, and the only application running at those sites was Citrix. The root cause of the performance degradation is still unknown, but bandwidth has been dismissed as a cause. We are currently evaluating WAN acceleration products as a possible solution.

Reducing the Time to Diagnose

Given that the length of time associated with diagnosing an application performance problem can be quite lengthy, we asked The Survey Respondents to indicate the average length of time it took to diagnose performance problems before and after purchasing the nGenius solution. Their responses are shown in Table 4.

The results displayed in table 4 are dramatic. For example, before deploying the solution it was rare that a performance problem was diagnosed in less than 1 hour. After deploying the solution, almost a third of all performance problems are diagnosed in less than 1 hour. Analogously, before deploying the solution, almost a third of all performance problems took more than 8 hours to diagnose. After deploying the solution, only five percent take that long.

Diagnosis Timeframe	Before Solution	After Solution
- 1 hour	2.8%	32.8%
+ 1 hour - 3 hours	24.3%	44.3%
+ 3 hours - 5 hours	20.8%	12.6%
+ 5 hours - 8 hours	19.4%	5.2%
+ 8 hour - 24 hours	14.6%	1.1%
+ 24 hours	18.1%	4.0%

Table 4
Length of time to diagnose a problem

The Manufacturing Specialist stated that before they deployed the nGenius System that his organization was often on the defensive against the common assumption that the network was the cause of any application degradation issue. He stated that right after they deployed the nGenius System that they were almost too successful with the use of the tool. In particular, now that they had the evidence to show that it was not the network and they often 'attacked back'. There has since been a culture change and his organization is now a lot smarter in terms of how they interact with the other IT organizations. As a result, the other IT organizations often calls them seeking their help resolving issues.

The Manufacturing Specialist noted that 10% of issues can take longer than a day to resolve and that some of them can go unresolved for months. He said that one problem they had recently involved the MAPI (Messaging Application Programming Interface) protocol. As it turns out, every 32 milliseconds the protocol would retransmit volumes of information. This took a long time to identify. He added that, "when something is fundamentally wrong, it can take a long time to identify and fix."

Solution Selection and Repair

In the case of fault management, there typically is no solution selection step. In particular, once it has been determined which component has failed, the solution is obvious: replace that component.

The Financial Engineer stated that effective problem resolution depends on having credible tools as well as people who work well together. He added that his organization tries to stay away from finger pointing. As an example, he commented that if they have an open trouble ticket, "It is not a network or a server issue. It is an IT problem and we all need to fix it."

The difficulty in selecting the solution to a performance issue can also be exemplified by an experience I had. I was recently hired by an IT organization that was hosting an application on the east coast of the United States that users from all over the world were accessing. Users of this application in the Pac Rim were complaining about unacceptable application performance. After some testing we determined that indeed the application performance was very bad and that the source of the problem was the content management tool that the application used. We also determined that some of the ways that the performance of the application could be improved included:

- Using a different content management tool
- Redesigning the WAN to create a shorter path from the users to the application
- Hosting an instance of the application in the Pac Rim
- Implementing network and application optimization techniques

Associated with each of the options were technical considerations such as how much would the option cost and how much improvement would it likely provide? Associated with each option were political considerations such as whose approval was needed to implement the option and who would pay for the solution? The combination of these options and considerations resulted in the period of time it took to select a solution being very lengthy.

The repair component of fault management differs somewhat from the repair component of application management. In the case of fault management, once you replace the defective part you fully expect the problem to be fixed. In the case of managing application performance, once you implement the chosen solution, you are less sure that the problem will go away. A good example of that phenomenon is contained in response #3 which is in the problem diagnosis section of this IT Impact Brief. As explained in that response, the IT organization is not sure of the root cause of the performance degradation. They are, however, contemplating deploying WAN acceleration products.

Response #3 is not unique. There are many situations in which it is difficult to determine with certainty the root cause of performance degradation and yet IT organizations are under pressure to take action. As a result, there is a reasonable chance that after implementing a solution that is intended to eliminate the application performance issue, that the IT organization will realize that the solution either did not work, or did not provide enough improvement. In many cases, this means that the IT organization has to repeat the problem diagnosis as well as the solution selection and repair processes.

Summary and Call to Action

As The Manufacturing Specialist stated, traditional fault management remains an important and a difficult task. And, as The Education Director pointed out, many organizations are under pressure to continually reduce their MTTR. With traditional fault management, the most difficult parts of troubleshooting are problem identification and diagnosis. As many of the interviewees pointed out, the key to successful problem identification and diagnosis is having effective tools.

While there are some similarities between fault management and application management, there are also some significant differences. For example, while fault management has been a traditional role of the network organization, managing application performance is a relatively new role. In addition, it is reasonable to have a 3 or 4 hour MTTR for fault management. However, as The Financial Engineer pointed out, when troubleshooting performance degradation, "The MTTR can get pretty large." Whereas solution selection and repair is not an important component of fault management, it is a very important component of application management.

The definition of MTTR changes when it is applied to troubleshooting performance degradation. The Financial Engineer provided some insight into how IT organizations should think about MTTR in this context. He stated that the measurement of MTTR begins when a user calls in and complains about the performance of an application. He added that the MTTR measurement ends when the user is no longer impacted or when the problem has been determined to be an issue with the application.

The Manufacturing Specialist pointed out that similar to the situation with fault management, having credible tools is a critical component of troubleshooting performance degradation. He added that other critical components include the IT organization's processes and culture. The Financial Engineer summarized the desired culture when he commented that if they have an open trouble ticket, "It is not a network or a server issue. It is an IT problem and we all need to fix it."