Application Performance Framework

A Guide to Performance Technology Selection

By Peter Sevcik, John Bartlett and Rebecca Wetzel

March 2004

With a burgeoning number of products and services dedicated to measuring and improving the performance of networked applications, it's challenging for managers to puzzle out which offerings best meet their needs. To help managers understand the many views of performance, and ensure application performance that meets business needs, NetForecast created The NetForecast Performance Framework. This report introduces The Framework and explains how it can be used to solve performance problems.

Application Delivery System

The application delivery system includes three components: a production element, a delivery element (centered around "the network"), and a consumption element. In the beginning the application delivery system was simple. Applications ran over consistent, uniformly performing wires controlled by a limited number of service providers. Things sailed along unchanged until client-server applications - which were really client-network-server applications - added some complexity to the system.

Then complexity snowballed. Applications began traversing many, ever more complex networks operated by scores of service providers. Today the network is big, complex, and important. To measure and improve performance of networked applications in this complex environment, vendors have introduced a "stew" of innovative technologies. More than 150 vendors claim to measure or improve the performance of networked applications. Each vendor brings its own view and approach to the table, with confusing, inconsistent, conflicting, and incomplete claims. Technologies ranging from caching, compression and load balancing - to protocol acceleration, rate shaping and QoS enforcement are competing for a place in your budget.

This highlights the need for The Framework (shown in Figure 1) to help enterprises make choices about application performance technologies.

		Asset Management			Experience Management		
		Provisioning	Efficiency	Protection	Accessibility	Quality	Safety
Real Time	Voice over IP						
	Video Conference						
Transactional	Terminal-Host						
	Client-Server						
	Web Applications						
	Web Services						
Stream	Streaming Audio						
	Streaming Video						
Bulk Data	Email						
	Peer-Peer						
	File Transfer						

Figure 1 – The NetForecast Performance Framework

NetForecast Report 5069

The Framework is a powerful tool for understanding performance needs and quickly identifying appropriate solutions. What follows is a description of The Framework [1] and a methodology enterprises can use to select vendors.

Framework Performance Functions

There are two types of performance attributes - *asset management*, which concerns the effectiveness of the assets that deliver the application - and *experience management*, which affects the user's experience with the application.

Asset management effectiveness is measured by the ease with which users are added to the service, and the maximum number of users that can be supported before resources are saturated and the application fails. These metrics revolve around system scaling or cost efficiency. Buying decisions for asset management technologies generally include a return on investment analysis.

Experience management effectiveness, on the other hand, is measured by the ability to make users happy and more productive. Assessing a new experience management technology focuses on benefits like increasing sales, lowering customer churn, increasing global reach, shortening time to market, and enabling real-time applications.

Although non-technical factors can affect asset and experience management, The NetForecast Framework only covers technology.

Asset Performance Functions

Asset performance is about *system* functions - provisioning, efficiency, and protection Asset management requires no user input - if all the boxes are working and connectivity exists, the system is operating properly and there's no need to ask the users if they agree.

Provisioning

Provisioning is the system's ability to establish new services, maintain current services or recover failed services. This function includes discovery, topology maps, alarms, uptime, routing stability, and fail-over. The system availability percentages that many management systems report are an example of provisioning metrics. Management systems generally determine whether service is available at the point where the application is generated – typically the data center. Effective provisioning requires that the application have high availability and recovery capabilities, made possible by such attributes as the ability to redirect users seamlessly to an alternate data center.

Efficiency

Efficiency is the system's ability to best utilize the assets that deliver the service. Metrics for efficiency include aggregate traffic, asset utilization, users per server, and users per Mbps. Cost metrics include capital and support expenses per user. Utilization measurements cover assets that belong to the enterprise, because it is valuable to ensure that money is not wasted on under-utilized assets. Value in this function can come from technologies that permit an asset (circuit, switch, server) to achieve higher utilization without collapsing (applications timeouts). Often these technologies improve the upper limit of the resource's practical operating range, and/or if there is more than one resource available, they improve efficiency by balancing the load among the resources.

Protection

Technologies that protect the system are in fact ensuring performance continuity. Protection describes the system's ability to protect itself from malicious or unauthorized use that would degrade the performance of the asset. Protection includes technologies that hide information or repel intruders such as firewalls, denial of service defenses, and

VPNs. Protection functions may be indirect in their protection, by authenticating users, identifying viruses, and/or improving the forensic capabilities of the IT staff.

Experience Performance Functions

Experience performance is about *application-specific* and/or *user-specific* functions, so any rating must relate to an end-user's point of view. A user group and application pair, like a remote office staff using a CRM application, define performance requirements. Therefore, talking to users is essential to any methodologies regarding these functions.

Accessibility

Accessibility is the system's ability to provide access to authorized users - the availability of the application *delivery* system between the edge of the data center and all the users' terminal devices. Accessibility is a measure of the impact on the user of issues like local access, user connectivity, and the effects of content filtering.

Quality

The quality function of The Framework covers the user's experience with the system. The metrics for this function are unique to each application class. One currently accepted measurement standard for quality is the Mean Opinion Score (MOS) for voice calls. The quality of a transaction application is measured by the user's task response time, that is the total time between a user entering a transaction on his device and a response that allows the user can proceed with his task. Although often used, simple measures of low-level protocol response times are weak indicators of total application response time.

Quality measurements must be interpreted properly to be useful. For example, a few measurements are not enough - and many measurements averaged together can be misleading because users experiencing frustrating performance are not noticed. (To learn more about NetForecast's definitions of satisfied, tolerating and frustrated performance, see references 2 & 3.)

Safety

Safety refers to the user's comfort level when interacting with the system. For example, safety is compromised when spam degrades the email experience, popups interfere with the application experience, viruses disable the user's machine, keystroke recording changes user behavior, and privacy concerns hinder the user's willingness to enter data.

Safety is different from asset protection described above because in the case of safety a user's privacy, identity, and/or comfort are being attacked or eroded with no additional risk to asset protection. In fact, asset functions may improve at the cost of decreased user safety. Safety risks can come from deliberate or accidental sources. Spyware is an example of a deliberate safety risk, and accidental risk could come from such sources as a poorly designed Web application that permits a user to mine cookie information to learn the identity and buying patterns of other customers of the site, thus potentially threatening other users who have visited the site [4].

Application Classes

Since The Framework is application specific, it is important to list the applications that are under consideration by each function. Some performance tools are narrow in scope and apply to only a few application types, while others cover all applications.

The Framework categorizes applications based on their traffic flow type - *Real Time*, *Transactional*, *Streaming*, and *Bulk Data*. Each traffic flow type requires a different approach to improve its performance.

Real Time traffic is time sensitive, and needs interactive responses. Time-sensitive packets must be delivered reliably and with low jitter, to ensure they are 'played' at the right time. Interactivity requires that overall network latency be low so responsiveness of the conversation or control loop is not lost. Voice and video conferencing, and remote control applications are typical of this category.

Transactional applications are dominated by many "turns," which are request/response pairs between parties [5]. The performance of transactional applications is often determined by the ability to complete transactions, rather than the time it takes to transfer data. This is especially true in high bandwidth, high latency environments, such as Web based services or remote server configurations.

Streaming traffic requires reliable packet delivery, and often requires substantial bandwidth – but does not have the same sensitivity to latency and jitter as real-time traffic. Streaming applications deliver audio or video information in a one-to-one, or one-to-many configuration, but only in one direction (outgoing).

Bulk Data applications move large blocks of data from one computer to another. Bulk Data applications do not have real-time requirements at the level of seconds, but they may have general performance requirements, such as completing a backup data transfer before the beginning of a business day. The primary performance issue for these applications is how to reliably move large quantities of data, as quickly as possible.

Let's look at the specific categories of applications in more detail.

Real Time Applications

Voice over IP (VoIP)
Voice traffic has two distinct components, the actual voice stream, and the control traffic needed to register, establish, and control calls. The control traffic behaves much like the interactive transactional applications described below, however, control traffic requires special treatment by the network because its performance is degraded by packet loss and jitter. Furthermore, because an interactive voice conversation has two participants who are expecting a certain level of interactivity, overall network latency must also be low. Bandwidth per voice stream is modest at 80Kbps or less. A VoIP call has two separate voice streams, one in each direction.

Video Conferencing
Video conferencing has the same characteristics as voice, with the addition of two streams of video information. Like voice streams, video streams suffer from loss and jitter, but they are different in the amount of bandwidth they consume. A video stream will use at least 300Kbps, and as much as 2Mbps. An interactive videoconference has two audio streams, and two video streams, one of each running in each direction.

Transactional Applications

Terminal-Host
Terminal-Host applications transfer minute amounts of data frequently. Terminal-Host includes 3270 type connections, as well as traditional terminals connected via a network terminal server.

Client-Server
Client-Server applications have a dedicated client installed on the desktop, which makes requests to a remote server for data or computational support. The client usually opens only one TCP connection to each server it uses, and these connections are open for an

extended period. Most enterprise applications from the 1980s and 1990s use this style of interaction, although many are now being converted to Web-based applications.

Web and Web Applications

This category includes Web browsing, as well as Web applications that use HTTP, Java or Active-X. These applications use a standard client - the browser. The browser opens many TCP connections to many servers to accomplish its task, but leaves connections open only for a short time. Browsers can often support four or more open connections concurrently. Web applications usually have a human at one end of the conversation, and a server at the other end.

Web Services

Web Services are contextual, in that they have knowledge about the objects being transferred, can identify objects by type, and perform actions appropriate to those object types. Web Services use XML and SOAP protocols. Web services traffic is multi-threaded and has short-lived connections. Web Services connections often have a computer on either end of the connection, with no human involved.

Streaming Applications

Streaming Audio

Streaming applications appear much like their real-time counterparts described above, and are just as susceptible to packet loss. However jitter and latency are not a problem for these streams, because these anomalies can be overcome with a buffer at the receiver. Streaming audio is usually unidirectional, from one source to one receiver or from one source to many receivers. Streaming audio may have a higher bandwidth than VoIP, since these streams often contain music, which requires more bandwidth to obtain better fidelity.

Streaming Video

Streaming video is like streaming audio except that video streams consume considerably more bandwidth than their audio counterparts.

Bulk Data

Email (SMTP)

This section refers to the back-end email task of moving emails from the SMTP sender to the PoP receiver - a task that can be executed in the background, or with lower levels of QoS in the network. The interaction between a PoP client and server, or between Outlook and an Exchange server, are Client-Server applications. This interaction takes all the emails from local clients, sets up a connection to the receiving PoP server, and transfers them. Email size varies widely because short text messages require very little data transfer, whereas emails with attachments can consume considerable bandwidth.

Peer-to-Peer

Peer-to-peer refers to Napster type applications that are typically transferring large files (e.g. music or video) from one client directly to another client. These files are not viewed in real-time, so the performance of the connection is not as important as its reliability

File Transfer

File Transfer applications are typically used for database replication or backup. The purpose is to move a large block of information reliably from one computer to another. Performance is not usually a concern at the seconds or microseconds level, but may be a concern at the macro level if there is insufficient bandwidth to complete the file transfer within its allotted time.

The Framework and the Application Delivery System

The intersection of the performance functions and application classes described above creates a matrix that is The Framework shown in Figure 1. The Framework is applied to the complex issues and pressing business needs of the application delivery system shown in Figure 2.

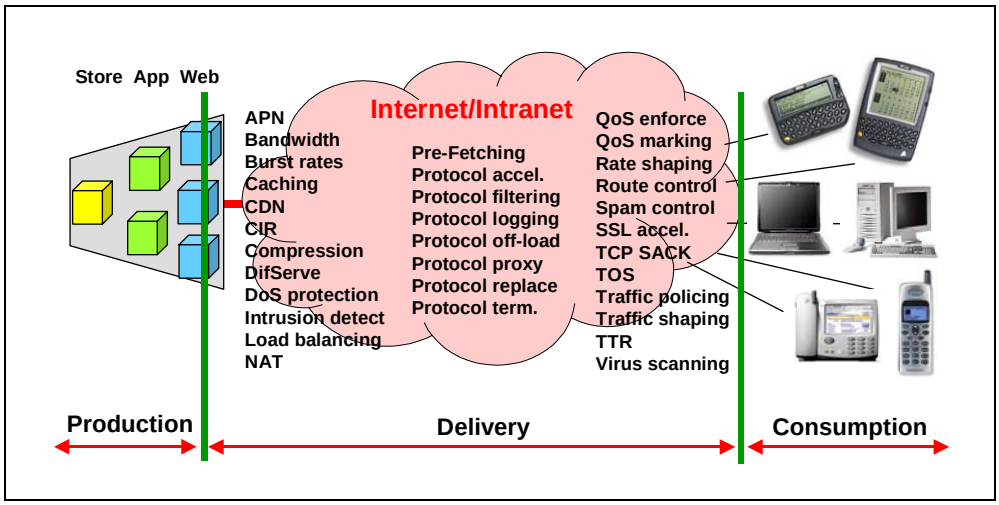


Figure 2 – Complex Application Delivery System

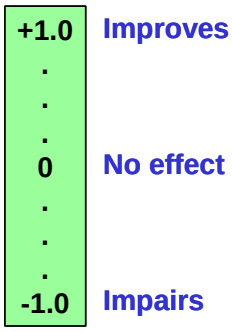
Many business applications are developed with a simplistic view of the delivery system, with the lion's share of attention heaped on the production side of an application. Yet many applications are complex, and need the coordination of Web, application, database, and storage servers to perform well. Some enterprises take the consumption side of the delivery system into account if users plan to access the application using a device with limited capacity like a phone. However, most enterprises and development teams assume that the two "goal posts" in Figure 2 are very close together.

Applications are often developed, tested, demonstrated, and approved in a LAN environment with a switched 100 Mbps delivery system. Imagine training a soccer team on a field that is only 50 yards long. Then the coach decides that the team is ready for competition and sends the team to play a match on a regulation-sized field 130 yards long! Many applications are deployed into environments where the goal posts are orders of magnitude farther apart than during development. No wonder so many IT strategies and application initiatives go splat upon deployment.

Fortunately, a growing number of technologies can be applied to counter the performance degradation seen in real world deployments. However, the specific benefit each technology brings varies by cell in The Framework. Therefore the strict performance definitions of each cell in The Framework are a way to sort and select the proper technology mix for a delivery system.

Using The Framework

The Framework must have a structured way to describe, rate, and evaluate technologies when it is used for technology selection decisions, This implies the need for uniform metrics across all of its cells.



Metrics of Performance

The values placed in the cells of The Framework represent the magnitude of the positive or negative impact any given solution applies to the characteristics and applications that lie at that intersection. The metric scale runs from –1.0 (very bad effect) to +1.0 (very good effect). For example, if a technology or vendor is given a metric of 0.8 at the intersection of Client-Server and Efficiency, it means that application of this technology will significantly improve the efficiency of the systems and network running a client-server application. If the same technology or vendor also receives a –0.3 at the intersection of Client-Server and Quality, it means that, although system efficiency was improved, the users' experience is degraded - albeit not as significantly as efficiency was increased. When matched to the requirements of the enterprise these tradeoffs help determine if this is an appropriate technology for their application.

To make comparisons across technologies, a consistent set of metrics must be applied to all solutions. Thus a +0.8 in efficiency applied to technology A should be roughly equivalent to a +0.8 in efficiency applied to technology B, even though the two technologies may increase efficiency in quite different ways. To this end, NetForecast has developed a set of statements that serve as guidelines when rating a technology or vendor, to insure consistency of the results. These statements are listed in Appendix A.

In general a technology that brings a single feature improvement to a framework intersection is given a +0.3 rank, and a –0.3 rank if the effect is negative. A technology that brings multiple features to a framework intersection ranks a +0.7 or +0.8, and if it brings significant improvement from multiple features, it gains a ranking of +1.0. Negative effects are rated in a similar way.

The Use Case

To be clear about the environment in which each technology is rated, a 'Use Case' must be written. The Use Case describes the environment in which the technology was considered and ranked. The Use Case is important, because decisions on the value that a new technology brings are dependent on the technologies already existing in the environment. Let's examine a case to understand this better.

Consider the data center manager who is investigating an SSL offload appliance to increase the efficiency of his server farm. His currently supported applications use SSL, but the SSL processing is being done by the servers themselves. By introducing an SSL appliance he hopes to free up the servers, allowing them to support more concurrent users.

SSL does provide a positive benefit both for the protection of the data center (Framework Protection column), as well as for the safety of the user's proprietary information (Framework Safety column). But in this Use Case, SSL is not being added to the environment, it is merely being moved. Hence both the Protection and Safety columns will get a 0 rating metric, while the column for Efficiency will get a positive metric because the servers have been offloaded.

The Use Case in this scenario is a description of the data center, with SSL currently implemented on the servers, and an assessment of how well a change to the delivery system will meet the objective of increased efficiency.

Case Study

In the following case study of a major European consumer electronics manufacturer, NetForecast used The Framework to assess and then improve the client's application

performance [6]. The client had developed three Web-based business-to-business applications supporting its supply chain encompassing some 300 suppliers worldwide. The system experienced very poor performance in the Asia-Pacific region, causing suppliers to complain, and in some cases take their business elsewhere. The lack of effective business process interaction with these suppliers was hurting their globalization strategy, and the threat of losing suppliers to competitors was a grave concern.

NetForecast interviewed key people within the organization to understand the situation. After presenting a tutorial on application performance to bring the decision makers to a uniform level of understanding, NetForecast asked managers to choose a weighting for the performance functions. The results were:

Provisioning	0.5
Efficiency	0.2
Protection	1.0
Accessibility	1.0
Quality	1.0
Safety	0.0

NetForecast then profiled the applications in use through a test account from NetForecast's headquarters. NetForecast then tested the behavior of the Internet from Asia-Pacific to the client's data center, using a small program installed in various business locations throughout the Asia-Pacific region. NetForecast analyzed the performance gains with each technology, some providing only small gains, others providing significant gains. The result of the investigation into various performance improvement technologies is summarized in Figure 3.

	Asset Management			Experience Management			Weighted Scores
	Provisioning	Efficiency	Protection	Accessibility	Quality	Safety	
Route Control							
Web-based SCM	0	+0.1	0	+0.1	0	0	+0.1
Content Delivery Network							
Web-based SCM	0	+0.3	0	-0.3	0	0	-0.2
Compression Software in Server							
Web-based SCM	0	+0.1	0	-0.1	+0.5	0	+0.4
Compression Appliance in Datacenter							
Web-based SCM	-0.1	+0.3	0	-0.1	+0.5	0	+0.3

Figure 3 – Example of Using The Framework

After a thorough review of technical alternatives, NetForecast recommended a compression product be installed at the data center, based on the positive results compression demonstrated in NetForecast analysis of the applications and technology alternatives. The client then had to make a decision about whether to buy the compression solution as software in the Web servers or as an appliance in the data center. The framework analysis in Figure 3 showed that software was a more appropriate solution for this client. The client implemented a simple software solution that had a dramatic positive effect on performance, and the overall business strategy.

How to Use The Framework in Your Enterprise

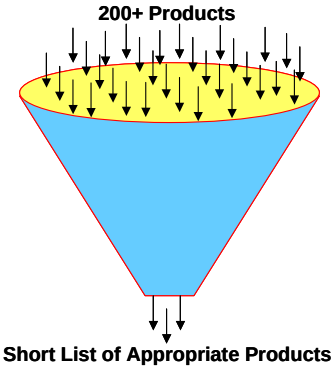
Here is the process we recommend when using The NetForecast Framework to determine the best performance solution for your needs.

Step One: Define your needs, taking into account the priorities of your enterprise. To do this, set up a team that includes users, business managers, and IT personnel. Work through the problems that need to be addressed (application performance, impact on the business, business direction, cost constraints, etc.), and identify your objectives. Is your objective to lower costs, increase revenues, increase user productivity or create a better experience? Use your objectives and the priorities of the organization to determine which of the six performance functions in The Framework (columns) are the most important to your organization.

Assign a relative weighting to each column that reflects their importance in meeting your goals. If the primary goal is to create a better user experience, the Quality column will get a heavier weighting than others. Conversely, if reducing cost is the goal, the Efficiency column gets more weight. Assigning relative weightings to each column will help sort out multiple vendor choices later in the process.

Step Two: Create a Use Case - a formal definition of the existing infrastructure, and the intended next step. The use case is important to how technologies are rated in The Framework, so it is essential to know if your use case matches the use case used when metrics were assigned to each technology.

What the Process Does



Step Three: Review available performance technologies. Technologies are assigned metrics in The Framework based on how they affect each intersection of the chart, and based on a specific Use Case. Use NetForecast's metrics, or create your own. Apply the weightings determined in Step 1 to the metrics for each technology, to determine which ones provide the most value given your situation. This step will quickly give you a short list of vendors who provide the best value.

Step Four: Interview the vendors on your short list, and question them about their technology to ensure you categorized them appropriately in The Framework.

Step Five: Choose the top two or three vendors from your short list, and pilot their technology to see first hand how their technology works with your application, to make sure the vendors are able to deliver the value they promise.

References

1. Sevcik, "A Framework for Enterprise Application Performance," BCR, November 2003
2. Sevcik, "Understanding How Users View Application Performance," BCR, July 2002
3. Sevcik, "How Fast is Fast Enough?" BCR, March 2003
4. Gomes, "Biggest Web Problem Isn't About Privacy, It's Sloppy Security," Wall Street Journal, January 26, 2004
5. Sevcik & Bartlett, "Understanding Web Performance," NF Report 5055, October 2001
6. Sevcik, "Accelerating E-Commerce," BCR, November 2002
7. Sevcik, "Web Performance – Not a Simple Number," BCR, January 2003

Authors

Peter Sevcik is President of NetForecast and a leading authority on network technology. He specializes in performance analysis and holds the patent on application response time prediction. He can be reached at peter@netforecast.com.

John Bartlett is Vice President of NetForecast, and has 24 years of experience designing at the chip, board, system, and network levels, with a focus on performance. He can be reached at john@netforecast.com.

Rebecca Wetzel is an associate in NetForecast with 20 year of experience in data networking. As a consultant, industry analyst and writer, she helps data networking vendors and service providers understand the needs of their target markets, and to market effectively to them. She can be reached at rebecca@netforecast.com.

Appendix – Metrics for Performance Enhancing Technologies

Asset Management - Provisioning

Improves
+0.1 Generally increases service availability (fewer boxes)
+0.3 Increases availability in a measurable way (simpler, faster to repair)
+0.7 Proactively restores lost service (redundancy)
+1.0 Guarantees service is available and operating within acceptable performance parameters with four-9s availability or better
Impairs
-0.1 Generally decreases service availability (more boxes)
-0.3 Causes service diagnostics to be less useful (increased complexity)
-0.7 Degrades MTBF and MTTR (complex, slower to repair)
-1.0 Significantly degrades service availability

Asset Management - Efficiency

Improves
+0.1 Increases the operating range of a single asset's utilization without impacting users
+0.3 Manages load across multiple components of an asset, uses spare capacity where available, broadens re-use of assets
+0.7 Manages system loading dynamically across multiple assets (e.g. circuits and servers)
+1.0 Provides significant (greater than 25%) savings across multiple resources
Impairs
-0.1 Specializes system components, making them less flexible in their application
-0.3 Specializes resources, increases uneven loading, reduces loading choices
-0.7 Forces uneven loading of resources, causes systems to manage tasks that it is inefficient at managing
-1.0 Forces over-provisioning or redundancy to insure service availability

Asset Management - Protection

Improves
+0.1 Improves the ability to detect and react to threats (smarter more reactive staff)
+0.3 Blocks one type of attack or vulnerability
+0.7 Protects against multiple attach types, provides verification of operation, has periodic update service to support learning of new threats
+1.0 Detects malicious attacks and manages them dynamically. Insures verification of process at the process level and within context. Dynamically defines baseline, and identifies abnormal behaviors.

Impairs
-0.1 Makes it harder to detect and react to threats (more staff time needed)
-0.3 Provides new openings for threats to enter the system
-0.7 Responds poorly to DoS attacks, opens new threat paths, bogs down system in face of attack
-1.0 Makes system vulnerable to malicious attacks. Compromises system's ability to validate users, opens new paths into system core.

Experience Management - Accessibility

Improves
+0.1 Increases the chances that more users get access via the same means (more diversity in access lines of the same type, e.g., dual homed, dial-up)
+0.3 Lowers barriers to users getting access (no software, no login, no account, or cookie based login, or integrated VPN)
+0.7 Expands the number of access methods (e.g., adding wireless)
+1.0 Actively polls or monitors user access attempts and failures, reports on specific access issues by region or other user class , supports proactive quality management

Impairs
-0.1 Generally reduces the visibility into detailed user access information, or increases access complexity
-0.3 Adds barriers to users getting access (requires plug-in, log-in, fast desktop, VPN)
-0.7 Impedes users ability to get access to service for portions of workday or in certain geographic locations
-1.0 A significant percentage of users can not gain access to the service

Experience Management - Quality

Improves
+0.1 Generally shortens the tail of response time distribution (faster 90th percentile)
+0.3 Moves more than 10% of the users to a better performance category on the NetForecast AppResponse Index [3,7], or MOS for voice
+0.7 Moves more than 20% of the users to a better performance category on the NetForecast AppResponse Index [3,7], or MOS for voice
+1.0 Moves more than 40% of the users to a better performance category on the NetForecast AppResponse Index [3,7], or MOS for voice

Impairs
-0.1 Generally lengthens the tail of response time distribution (slower 90th percentile)
-0.3 Moves more than 10% of the users to a worse performance category on the NetForecast AppResponse Index [3,7], or MOS for voice
-0.7 Moves more than 20% of the users to a worse performance category on the NetForecast AppResponse Index [3,7], or MOS for voice
-1.0 Moves more than 40% of the users to a worse performance category on the NetForecast AppResponse Index [3,7], or MOS for voice

Experience Management - Safety

Improves

+0.1 Able to prevent some safety issues from occurring (pop-ups, spam, viruses, etc.)
+0.3 Adds assurances that user privacy will not be compromised (via cookie mining, etc)
+0.7 Able to totally prevent one category of safety issue, and can impact other categories
+1.0 Able to totally prevent all known safety issues, has periodic update capability to keep protection current (virus, identity theft, spam, spyware, cookie, popup, anonymous)

Impairs

-0.1 Adds significant distraction to the work environment
-0.3 Causes safety holes to be opened, may disclose information or causes leaks of personal or work-related information to undisclosed parties
-0.7 Actively monitors user's activity without user's knowledge
-1.0 Destroys or counteracts user safety mechanisms in this environment