

Taking Control of Secure Application Delivery

By Jim Metzler



Introduction 3

The Changing Application Delivery Model4

The Need for an Integrated Solution4

The Building Blocks of Network and Application Optimization5

 Data Compression7

 Differential Compression8

 Block Level Data Compression8

 Object Caching8

 Bandwidth Management9

 Protocol Optimization10

 Server and Client Offload10

An Integrated Architecture11

 Importance of Accelerating and Securing Web Applications11

 Converged Acceleration Architecture12

Summary and Call to Action12

Introduction

It has been well documented in many publications that if a company's CIO has a recent technical background it is usually in the arena of applications. It is also widely understood that a company's business and functional managers care much more about the applications that they use on a regular basis than they do about any component of the IT infrastructure. These managers either set or heavily influence the IT budget, and IT organizations should therefore ensure their satisfaction. Ensuring that these managers perceive that they get value from IT in general, and from the IT infrastructure in particular is of paramount consideration for any IT organization. To achieve this goal, IT organizations must be able to relate the value of the IT infrastructure to the applications with which these senior managers are concerned.

One of the best ways to relate the value of the IT infrastructure to senior management is by ensuring the appropriate performance of the applications that these managers use on a regular basis. One factor making the task of ensuring application performance a bit easier is the large and growing array of application acceleration techniques that are available in the marketplace. Having such a wide array of possible techniques, however, also complicates the very task of ensuring application performance. The breadth of these techniques creates questions in the minds of IT organizations about the applicability of each.

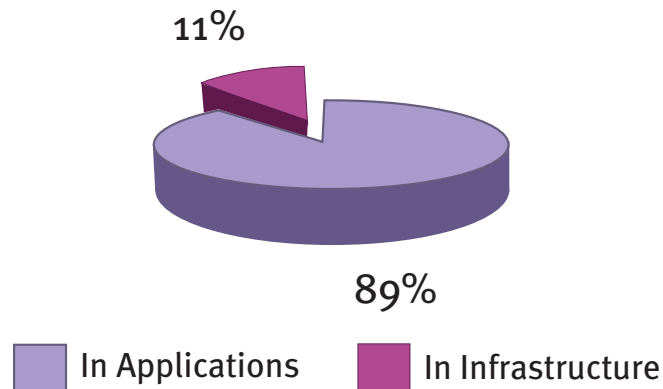
Another complication is that the model for how applications are delivered is rapidly changing. As recently as a few years ago the 80/20 model of application delivery stated that eighty percent of a company's employees accessed corporate applications locally. The new 80/20 rule states that at any given time, eighty percent of a company's employees access corporate applications remotely.

Security is another factor complicating the task of ensuring appropriate application performance. In particular, there is growing use of SSL (Secure Socket Layer). SSL is designed to enable secure communications over the Internet in part by encrypting the data. Because SSL encrypts the data, however, it is impossible for some of the existing network and application acceleration techniques to function correctly.

The goal of this white paper is to present an architectural framework that IT organizations can customize for use in their environment. In order to achieve that goal, the white paper first details how the traditional application delivery model is changing. It further identifies the type of control that IT organizations must have in order to successfully support this new model. The white paper then describes some of the most important application acceleration techniques and presents the applicability of each. The white paper concludes with a description of an architecture that enables both optimal application performance and security.

The Changing Application Delivery Model

As previously noted, it is widely understood that a company's senior management team finds more value in applications than they do in the IT infrastructure. To quantify this bias, Ashton, Metzler & Associates recently asked approximately 200 IT professionals to identify where their company's business and functional management saw value from the IT function. As shown in Figure 1, the response to that question was quite dramatic and underscores the need for IT organizations to be able to relate the value of the IT infrastructure to the company's key applications:



Source: Ashton, Metzler & Associates
Relative Value of IT
Figure 1

The task of relating the value of the IT infrastructure in general, and of the WAN in particular, to the applications that matter most can be daunting. One of these challenges stems from the fact that the vast majority of a company's employees no longer reside in a headquarters site where they access corporate applications over a LAN. These employees need to access to the full range of corporate applications from virtually anywhere, including the company's branch offices, the employee's home, a customer site, a hotel, or an airport.

It could be argued that just because the WAN enables a company's remote employees to access corporate applications that it provides demonstrable business value. However, the use of a WAN to access corporate applications often has a negative impact in terms of how the applications perform. In particular, applications that run well over a LAN often run poorly over a WAN. This is one of a number of instances in which the WAN is more likely to be perceived as an impediment to achieving a business objective, rather than as an enabler.

To understand why the introduction of the WAN makes such a difference, consider the case of an employee located at headquarters accessing an application locally. This LAN typically has a capacity of between 100 Mbps and 10 Gbps.

The distance between the employee and the application is also usually one thousand feet or less. This combination of high capacity and limited distance results in a network

that can support huge amounts of traffic and that has negligible delay, jitter, and packet loss.

Consider the contrasting case of an employee who needs to access an application remotely. The information transfer between the employee and the application must transit a WAN prior to transiting the headquarters LAN. In the vast majority of cases, the capacity of the WAN is less than one tenth of one percent of the capacity of the headquarters LAN - approximately 56 Kpbs to 1.5 Mbps. As a result, the WAN can only support a relatively small amount of traffic.

Also, instead of being one thousand feet away from the application the employee may be hundreds or even thousands of miles away from the application. This combination of long distance and shared WAN resources typically results in an approximate 50 ms delay in the WAN. A delay of this magnitude can have a disastrous affect if the company is running a chatty protocol such as CIFS (Common Internet File System) over the WAN.

Another challenging aspect of the changing application delivery model is the ongoing movement to consolidate servers out of branch offices and into centralized data centers. Part of the motivation to consolidate servers is that it reduces cost. It lessens the need for servers as well as the associated software licenses, real estate, and IT resources that are required to support and manage the infrastructure. Consolidating servers also makes it significantly easier to respond to the ongoing enactment of government and industry regulations. Such regulations as the Sarbanes-Oxley Act require companies to put a greater emphasis on ensuring the accuracy, security, and confidentiality of data.

As was the case with supporting remote employees, it could be argued that the WAN provides business value because it allows employees in branch offices to access centralized servers. In many cases, however, having remote users access key Microsoft applications over a WAN results in unacceptable performance. In particular, common functions such as opening a file take only milliseconds when the client and the server are in the same branch office. Such operations can take several seconds when the client is in a branch office and the server is hundreds or thousands of miles away in a data center. This added delay is yet another reason the WAN is likely regarded as an impediment to achieving the business objective rather than as an enabler.

The reason that server consolidation often results in significant extra delay is that Microsoft file services rely on the CIFS protocol, which is chatty. Chatty protocols, which were designed to run over a LAN, typically exchange tens or even hundreds of messages between sender and receiver for each transaction. In addition to CIFS, other examples of chatty protocols include HTTP (HyperText Transport Protocol), MAPI (Messaging Application Programming Interface), MMS (Microsoft Media Server protocol) and RTSP (Real Time Streaming Protocol).

CIFS decomposes all files into smaller blocks prior to transmitting them. Assume, for example, that a client is attempting to open a large file on a remote server. CIFS would decompose that file into tens, or possibly hundreds of small data blocks. The server sends each of these data blocks to the client where it is verified and an acknowledgement is sent back to the server. The server must wait for an acknowledgement prior to sending the next data block.

To quantify the delay, assume that CIFS decomposes a file into one hundred data blocks. Further assume that the only delay in the WAN is a one-way delay of 50 ms. Because one hundred round trips are required to open the file, it would take ten seconds to open the file. Few users would find this to be acceptable.

A further challenge of the changing application delivery model is that the use of applications no longer stays within the boundaries of a company. In the typical client-server application, for example, instrumenting both the client and the server provides the IT organization a significant amount of management information and control. In a growing number of cases, however, a company's employees are accessing a hosted application such as Salesforce.com. In these situations, the IT organization can no longer instrument both ends of the application.

A related situation is that the majority of companies have begun to move to a Web services approach to application development for applications such as Supply Chain Management and Customer Relationship Management. In the majority of cases, the application is comprised of Web services that reside in data centers that belong to multiple organizations. Again, this means that the IT organization can no longer instrument all of the components of the application.

The Need for an Integrated Solution

As suggested in the introduction, in the majority of cases, senior management tends not to recognize that the company's WAN provides any significant business value. For this reason it is incumbent on the IT organization to demonstrate to senior managers the ability of the WAN to improve the performance of the applications that they use on a regular basis.

To successfully support the emerging application delivery model, IT organizations must ensure

- The appropriate performance of applications as seen by remote workers
- That viruses, worms, trojans and spyware are not allowed onto the branch or the remote office networks
- That employees are not making inappropriate use of the Internet

Just as the application delivery model is changing, the network and application optimization model must also change. In particular, the current network and application optimization model requires IT organizations to implement appliances in each branch office. In many cases these highly specialized appliances sit in branch offices next to numerous other appliances, each of which is performing some other highly specialized task in order to make individual applications run better over the WAN. Because of the complexity involved, few companies want to deploy a network and application optimization model that results in having their branch offices littered with appliances.

The emerging network and application support model must be simpler than the current support model. What is needed is a network and application support model that is built on an integrated management system. This model must be able to provide key functions such as those described in the following section.

The Building Blocks of Network and Application Optimization

The goal of network and application optimization is to ensure appropriate application performance. At a conceptual level, there are five primary components of network and application acceleration. They are to

- Reduce the amount of data that is sent over the WAN
- Prioritize traffic that is business critical and delay sensitive
- Ensure that the WAN link is never idle if there is data to send
- Reduce the number of round trips, or application turns that are necessary for a given transaction
- Offload computationally intensive tasks from client systems and servers

This section of the white paper describes some of the specific techniques that are used to accomplish the goals listed in the preceding list. Please note that it is possible to deploy these techniques in isolation or in concert with other techniques. You could, for example, deploy only data compression. It is also possible, however, to deploy data compression and bandwidth management. Also, please note that any reference that quantifies how much benefit is associated with each of the techniques is intended only to provide insight. In particular, the benefit that is associated with each of the techniques described in this section is highly dependent on the specific traffic patterns of the company that implements the techniques.

Data Compression

The goal of all compression techniques is to reduce the amount of data that must be transmitted and stored. As such, implementing compression has the affect of a one-time increase in the capacity of the communication channel or storage resources.

Data compression techniques can be used to reduce the size of the transmitted file without requiring the receiver to have an earlier version of the transmitted file. Data compression algorithms find redundancy in a data stream and use encoding techniques to remove the redundancy, creating a smaller file. A number of familiar lossless compression tools for binary data are based on Lempel-Ziv (LZ) compression. This includes zip, PKZIP and gzip algorithms.

HTTP/1.1 supports compression with gzip, compress, or deflate/zlib, all of which are LZ based. LZ develops a codebook or dictionary as it processes the data stream and builds 12 bit codes corresponding to sequences of data. Repeated occurrences of the sequences of data are replaced with the 12 bit codes. The LZ codebook is optimized for each specific data stream and the decoding program extracts the codebook directly from the compressed data stream.

With LZ based compression tools, the degree of compression depends on the size of the input and the distribution of common substrings of data. Typically, text files, such as source code or English, can be reduced in size by as much as 60-70%. For other types of data with many possible data values (such as high bit-depth color images), LZ may prove to be quite ineffective because repeated sequences are fairly uncommon. As a result, audio and video applications generally avoid LZ based compression tools and instead use special purpose codecs, such as MPEG and JPEG compression algorithms.

Differential Compression

The goal of differencing algorithms is to avoid sending an entire file from origin to destination. In particular, the goal of differencing algorithms is to only send the changes that have been made to the file since the last time it was sent.

To achieve this goal, differencing algorithms describe a new version of a file as a set of changes to a previous version of the same file, where the previous version of the file is often referred to as the base version. All binary differencing algorithms partition a file into two classes of variable length byte strings: those strings that appear in the base version and those that are unique to the version being encoded. The latter strings are included in what is often referred to as a delta file. The delta file contains the minimum set of changes that the receiver at the destination needs in order to build the new version of the file.

One common example of the use of differential compression is to minimize the download size of operating system and other software updates. In such cases, the size of the delta file is only a small percentage of the base version of the file. As such, differential compression can greatly reduce bandwidth requirements for software distribution, replication of distributed file systems, and file system backup and restore.

While differential compression is constrained to those cases where the receiver has stored an earlier version of the file, the degree of compression is very high; 10x or higher is not uncommon. Also, as a final note, the processing needed to create the delta file can often be performed well in advance of data transmission.

Block Level Data Compression

Data compression was previously discussed in the context of compressing entire files. The same basic algorithms can also be applied to individual blocks of data, which—taken—together would comprise a larger file or object. The LZ-based algorithms are inherently block oriented, so they are readily modified to consider each block of input data (up to 32 KB or 64 KB for example) as a separate data entity. Block-level data compression lends itself to those applications that send streams of data over the network, especially where individual blocks may represent quite different types of data.

Block-level data compression has a significant advantage: it can be applied to data transfers of any TCP- or UDP-based application that incorporates error recovery at the application level. Working at the block level introduces only a small amount of latency because the receiver does not need to store and process the entire file before decompression can begin. Usually, the degree of compression is quite dependent on the length of the block, as well as the frequency of repetitive strings in the block.

Object Caching

Similar to compression, the goal of caching is to reduce the amount of data that must be transmitted. Object caching achieves this goal by storing copies of remote application objects in a local cache, which is generally on the same LAN as the requesting system. With object caching, the cache server acts as a proxy for a remote application server, an

FTP server or a Web server, for example. Object caching is therefore limited to those applications and protocols supported by specific proxies.

The most common example of object caching is a Web cache appliance that serves as a proxy Web server, caching frequently accessed objects from Web pages that are hosted remotely over the Enterprise WAN or on the Internet. Client browsers are configured to connect to the proxy server rather than directly to the remote server. When the request for a remote object is made, the local cache is queried first. If the cache contains a current version of the object, the request can be satisfied locally at LAN speed and with minimal latency. Most of the latency involved in a cache hit results from the cache querying the remote source server to ensure that the cached object is still current.

If the cache does not contain a current version of the object, that object must be fetched by the cache, cached, and then forwarded to the requestor. Loading the object into the cache could potentially be facilitated by one of the other acceleration techniques described in this section, such as data compression. By deploying Web caches, an organization will typically reduce its HTTP traffic by 30% to 50%.

Object caching is an area in which application acceleration vendors can add value with innovations that further minimize latency and maximize the frequency of cache hits. The Web cache, for example, may use its spare cycles to ensure that frequently accessed dynamic objects are continuously refreshed to increase the frequency of cache hits. Another optimization technique is to request multiple objects per request to the source server. This technique can accelerate the refresh time by fetching objects in parallel and exploit the full computational resources of sites with clusters of load-balanced servers.

Bandwidth Management

The other techniques that were described in this section serve to increase the efficiency of the WAN. Even if the WAN is made extremely efficient, however, there are times when large volumes of traffic result in WAN congestion and hence WAN latency. The goal of bandwidth management, therefore, is to prioritize traffic that is latency-sensitive and business critical.

One example of a latency-sensitive, business-critical application is VoIP. Over the last few years the majority of companies have made at least some deployment of VoIP. One of the features that distinguish VoIP from a more typical data application is the rigorous demand that voice places on the underlying IP network. The ITU (International Telecommunication Union), for example, recommends that the end-to-end delay associated with a voice call not exceed 150 ms. Experience has shown that it is possible to exceed that goal by a small amount. If the delay becomes too large, however, the quality of the voice call degrades noticeably.

Another example of a latency-sensitive, business-critical application is SAP®. Several SAP modules are notably delay sensitive. An example of this is the Sales and Distribution (SD) module, used for sales order entry. If the SD component is running slowly, a company can compute the lost productivity of the company's sales organization as they waste time waiting for the SD module to respond. Furthermore, if

the SD module times out, this can irritate the customer, causing them to take their business elsewhere.

The role of bandwidth management is to give latency-sensitive, business-critical applications priority over other applications. As a minimum, bandwidth management must be able to give VoIP traffic priority over an application such as SAP, or be able to limit the amount of bandwidth associated with a given port, such as port 80. It is also very important to be able to apply bandwidth management more finely. For example, a given company might want to have the bulk of their VoIP traffic have priority over SAP traffic. That company might, however, also want to have the SAP traffic of certain key users to have priority over any other traffic, including VoIP.

Protocol Optimization

As previously mentioned, Microsoft file services rely on the CIFS protocol, which decomposes all files into a large number of small data blocks prior to transmitting them one at a time from the server to the client. This results in a situation in which the transmission link alternates between being busy when the server is sending a data block, and being idle while the server waits for an acknowledgement of that data block from the client.

One key step in optimizing the performance of CIFS is to consolidate a number of data blocks into a larger file. This consolidation improves CIFS performance in two significant ways. The first way is that it ensures that the WAN link is seldom idle if there is data to send. The second way is that it eliminates un-necessary round trips.

Using both block level compression and differential compression can further optimize CIFS. Assume, for example, that a user in a given office has downloaded a file. The next time that anyone in that office wants to download the file, block-level compression can be applied just to those blocks that have changed since the last time the file was sent.

Server and Client Offload

This component of network and application optimization offloads server and client processing tasks that are network related and that consume considerable CPU cycles. Examples of these types of tasks include compression and decompression, protocol optimization, intrusion detection, and virus scanning.

As a result of offloading these tasks, more computing capacity is available for application processing on both servers and clients. Offloading processing to a shared resource is especially desirable when the same computationally intensive process is replicated at numerous servers or desktops. A second benefit of offloading processing from the end systems to a shared resource is that it reduces the operational complexity of software suites in both the data center servers and in the desktop systems. Offloading of network-related processing from clients and servers is the basic rationale for running most application acceleration functions on dedicated appliances or integrated platforms. A primary example of server offload is the SSL processing in the data center. SSL offload allows the intranet Web and Internet eCommerce servers to

process more requests for content and handle more transactions. This technique provides a significant increase in the performance of these secure sites without adding more server capacity.

Another data center server offload strategy that can help to accelerate applications is based on TCP Offload Engines (TOE) in the server NICs. With TOE, the entire TCP/IP processing for each network session is offloaded from the host CPU to a specialized processor in the TOE NIC. TOE NICs have been able to dramatically reduce CPU loading and application latency for Gigabit Ethernet-attached servers.

An Integrated Architecture

There are a number of approaches to network and application optimization that are available in the marketplace. This section of the white paper addresses some of the more critical issues that enterprises face in deploying these approaches. It also offers a high-level architecture for enterprise-wide network and application optimization.

This architecture is predicated on the belief that deploying a different appliance for each acceleration technique is undesirable for two reasons. One reason is that having a separate device for each acceleration technique would place a significant burden on the network management and operational staff. The second reason is that having a separate device for each acceleration technique makes it extremely difficult, if not impossible, to achieve any synergies between the various techniques. Therefore, what is needed is an architecture that integrates most if not all of the desired optimization techniques.

Importance of Accelerating and Securing Web Applications

For most enterprises considering network and application optimization solutions, Web traffic is likely to be a major concern. In the first place, HTTP traffic generally consumes most of the Internet access bandwidth. Second, Web-based enterprise applications and Web front ends for legacy enterprise applications are driving rapid growth in secure Web intranet traffic based on HTTPS (HTTP encrypted with SSL). For many organizations HTTPS accounts for as much as 30% of the total WAN traffic. In the future, increases in HTTPS traffic will be accelerated by the growing acceptance of Web services-based applications.

Therefore, a network and application optimization solution should support Web proxy functions. This increased support should include a combination of HTTP object caching and HTTPS/SSL protocol processing, including both session termination and origination. HTTPS origination allows the Web proxy to originate secure requests to the source content server. Therefore, when a client browser accesses a secure Web site on the Internet, two SSL sessions are created: one between the client and the proxy and another between the proxy and the source server. HTTPS termination allows the proxy to terminate intranet or Internet HTTPS and offload SSL processing from a co-located origin server. Doing so increases the capacity of a Web site or Web-based enterprise applications without adding more servers.

In addition to accelerating both HTTP and HTTPS applications, the proxy's ability to parse SSL traffic allows it to control rogue Internet applications. Some applications attempt to use SSL encryption as a means of circumventing bandwidth management solutions and security measures that examine session content. Rogue applications that sometimes exploit SSL include IMS, P2P file sharing, and peer-to-peer VoIP.

Proxies that are not capable of SSL processing must pass all SSL traffic between enterprise clients and the Internet on TCP port 443 without accelerating it or examining the content. This raises the possibility that rogue applications or even Web e-mail may introduce viruses, worms, spyware and other malicious or unwanted content onto the enterprise network.

Converged Acceleration Architecture

Figure 2 shows a high-level architectural view of how a multi-functional network and application acceleration platform, including Web proxy functions, can be deployed in the enterprise network. The vendors that provide solutions to implement these functions use a wide range of terms to describe their network and application acceleration platforms. For simplicity and vendor neutrality, network and application acceleration platforms will be referred to in this section as accelerators.

The following list presents a brief description of each of the key components of this architecture. Note that each of the key components is numbered and this number is used in Figure 2 to indicate where that component belongs in the architecture.

- 1. Accelerators in the enterprise remote offices are paired with an accelerator attached to the central site WAN router.** This solution provides functions such as data compression and caching, which can minimize remote client response time and WAN bandwidth consumption for a wide range of applications.
- 2. An accelerator behind the Internet firewall/IPS.** This solution provides the Web proxy and SSL session origination that minimizes Internet access traffic. It also accelerates, controls, and secures local client access to Internet resources.
- 3. An accelerator in the data center.** This solution provides the Web proxy and SSL session termination that accelerates Web-based enterprise applications. It also off-loads SSL processing from Web servers dedicated to enterprise applications.
- 4. An accelerator in the DMZ.** This solution can provide SSL session offload for enterprise Web servers being accessed over the Internet by the enterprise's customers and partners. If load balancers or Layer 4-7 switches are employed to front end the Web servers in the DMZ, these devices may support the required SSL termination functions, obviating the need for a separate Web proxy/SSL platform.

Summary and Call to Action

As described in the introduction, the senior management at most companies do not believe that the company's WAN provides much business value. In order to show the business value of the WAN, IT organizations must demonstrate the value that the WAN offers to the applications upon which these senior managers depend.

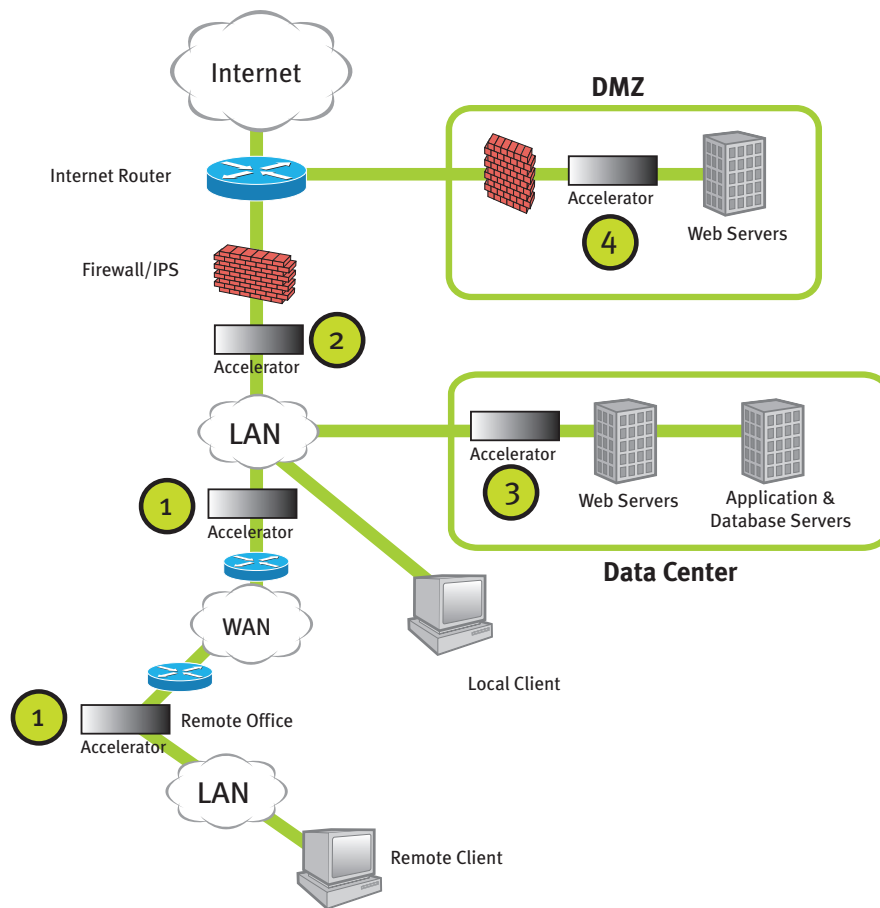


Figure 2
Application & Acceleration Architecture

Complicating the task of demonstrating the business value of the WAN is the fact the traditional applications delivery model is changing. One trend driving this change is that the majority of users once accessed applications locally, over a high-speed, low-latency LAN. Now, however, they access applications over a relatively low-speed, high latency WAN. Furthermore, many companies are consolidating servers out of branch offices and into centralized data centers. The combination of these two trends often results in poor application performance, which makes it appear as if the WAN is an impediment to a company achieving its business goals.

To ensure appropriate application performance, IT organizations have begun to deploy network and application functions that accomplish one or more of the following tasks:

- Reduce the amount of data that is sent over the WAN
- Prioritize traffic that is business critical and delay sensitive
- Ensure that the WAN link is never idle if there is data to send
- Reduce the number of round trips, or application turns that are necessary for a given transaction
- Offload computationally intensive tasks from client systems and servers

As described earlier, there is a very wide range of techniques that can be used to implement these functions. IT organizations will not be successful, however, if they litter their offices with various appliances each intended just to implement

one of these niche technique. In order to be successful, IT organizations must deploy a solution that can support key techniques, such as the ones that were described in earlier.

The growing importance of security and the use of secure protocols such as HTTPS and SSL also means that IT organizations must develop an application acceleration architecture. Such architectures (depicted in section 5 figure 1) specify where security and application acceleration functions should be deployed. One of the primary goals of this architecture is to ensure that a company does not have to choose between security and application performance.

When IT organizations are choosing the solution they will use to implement such an architecture, they should consider traditional selection criteria. Such considerations include the breadth and extensibility of the solution as well as the performance, cost and reliability of the solution.

There are, however, other selection criteria that IT organization should also consider. These include the

- Breadth of applications that the solution supports (for example, Web based applications, Microsoft Exchange, collaboration)
- Breadth of techniques that the solution supports
- Ability of the solution to support policy management and hence ease the burden of configuration management
- Operational ease of implementing and supporting the solution

An architecture that seamlessly integrates security with network and application optimization combined with a solution that satisfies the suggested criteria is paramount. Such architectures enable IT organizations to ensure the secure and effective performance of the company's key applications.

