How to Approach Application Delivery

March 2007

1.0 Introduction

As recently as a few years ago, application delivery was not an important topic for most IT organizations. Over the last few years, however, this has changed dramatically. Currently application delivery is an important topic for most IT organizations for two fundamental reasons. One of these fundamental reasons is that companies are increasingly implementing key business processes using applications that run on their IT infrastructure. If these applications are not performing well, it means that those business processes are not running well. In particular, if these applications are not running well it could result in a company being unable to respond to a customer inquiry, ship product, or close their books at the end of the quarter.

The second fundamental reason why application delivery has become so important is that the task of ensuring acceptable application performance is complex, and is becoming increasingly more complex over time. One of the factors that make application delivery so complex is that many components of IT impact application performance. These components include the applications themselves, the network (LAN, WAN and SAN), the servers, the storage, the operating system, the backend database, as well as the varying security mechanisms that IT organizations have implemented.

Another factor that makes application delivery complex is the contrasting dynamic that is driving the location of a company's employees and their IT resources. In particular, a few years ago the majority of a company's employees worked in a headquarters facility. Today, driven in part by the need to be close to customers, the majority of employees work from home or in a branch office. In contrast, driven by the need to reduce cost and get better control over their company's data, most IT organizations are centralizing their IT resources. For example, many companies are taking applications, servers, and storage out of branch offices and placing them into centralized data centers. At the same time, the majority of large companies are also reducing the number of data centers that they support and moving to a single hosting model for many of their applications.

When the majority of employees resided in a headquarters facility they typically accessed applications over a high-speed, low latency LAN. The combination of the movement to distribute employees and centralize IT resources has changed that. Now the majority of employees work in places other than headquarters and access centralized applications over a relatively low-speed, high latency WAN. In addition, the movement to reduce the number of data centers and implement a single hosting model means that the WAN links between the branch office and the data center are getting longer. Increasing the length of WAN links results in increased WAN latency and increased potential for network congestion.

Given that most IT organizations have just recently began to focus on application delivery, it is rare to find an IT organization that has well-developed, effective processes

for ensuring acceptable application delivery. The goal of this white paper is to help IT organizations develop those processes.

With that goal in mind, this white paper will briefly describe the different classes of applications that transit the WAN and will also illustrate the fact that the factors that cause one class of application to perform badly are not necessarily the factors that cause another class of application to perform badly. The white paper will also describe some of the primary WAN optimization technologies and will identify some steps that IT organizations should take in order to create effective application delivery processes.

2.0 Applications that transit the WAN

While not mentioned in the introduction, another factor that complicates application delivery is the number and types of applications that transit the corporate WAN. For example, the typical enterprise has tens and often hundreds of applications that transit the WAN. These applications can be categorized based on the:

- Mission-critical nature of the application
- Demands that the application makes of the network
- Functionality provided by the application

Mission-Critical Applications

While a company may well have hundreds of applications that transit the corporate WAN, in most cases that company typically runs the bulk of its key business processes utilizing a handful of these applications. These applications will be referred to in this white paper as being mission-critical.

Application Demands

Since they make different demands on the network, another way to classify applications is whether the application is real-time, transactional or data transfer in orientation. One of the primary factors that distinguish these three classes of applications is the their delay characteristics. For example, the delay associated with real-time applications is measured in milliseconds, while the delay associated with transactional applications is measured in seconds and the delay associated with data transfer-based applications is measured in minutes and hours.

Application Functionality

Another way to classify applications is based on the functionality that they provide. For example, some applications support collaboration while others support real time transactions. The next section of this white paper will identify a number of classes of applications based on the functionality that they provide. In addition, the next section will describe some of the factors that cause each class of application to perform poorly.

3.0 Application Issues

Each class of application may be characterized in terms of its requirements for bandwidth and the user issues that are caused by excessive network latency, jitter and packet loss. Understanding the significant differences among the various classes of WAN application is an essential step in optimizing application delivery. The remainder of this section provides an overview of the performance issues that may be encountered for each class of application.

Convergence

Applications in this category include VoIP, audio and video conferencing, and streaming video. VoIP and audio conferencing have low bandwidth per call requirements but are sensitive to the latency, jitter, and packet loss that can result from congestion or over-subscription of bandwidth guarantees. Video is somewhat less sensitive to jitter and packet loss but can consume several hundred Kbps of bandwidth per endpoint. These real-time applications are implemented over UDP and cannot throttle back bandwidth utilization in response to network congestion. Where convergence applications are supported over the WAN, QoS (Quality of Service) functionality must be exploited to ensure that bandwidth is shared in accordance with enterprise policies.

Real-Time Transaction Processing

These interactive applications include the transactional processing aspects of ERP, CRM, and other enterprise applications. This type of application is typically not sensitive to jitter or to moderate amounts of latency. However, these applications are sensitive to congestion and, in particular, to any resulting packet loss. Packet loss is important because when packets are lost the end system's TCP windowing mechanism seeks to reduce network congestion by throttling back its demand for bandwidth through the reduction of the TCP window size. As less bandwidth is used, response times can easily grow to where users complain. If the congestion is caused primarily by non-TCP applications that have no throttling capability, the transaction processing end systems will be forced to keep the window size small for an extended period of time.

Collaboration

Collaboration applications include client/server file access and file sharing via Microsoft CIFS and NFS, and sharing of large files as email attachments. CIFS and NFS are chatty protocols[1] that were designed primarily for a low-latency LAN environment. NFS and CIFS are based on Remote Procedure Calls (RPCs) that are executed sequentially, greatly magnifying the effects of end-to-end latency and resulting in poor response times. For example, with CIFS, accessing a 1 MB file can involve 120 RPCs, which means that the user response time has to exceed 120 times the end-to-end network latency. When

[1] Chatty protocols typically require tens or even hundreds of application turns to complete a single transaction.

NFS and CIFS are run over UDP, they can contribute to the bandwidth starvation of TCP-based flows.

Server Centralization

As enterprises seek to control costs by consolidating network services (including SMS, DNS, and DHCP) at central sites, remote sites are forced to access these services over the WAN. While the services themselves are not particularly sensitive to network performance, they are usually given fairly high priority because of their fundamental role in network operations. As a result, these services can compete for bandwidth with mission critical enterprise applications sharing the same QoS traffic class.

Bulk Transfers

Applications in this category include storage replication, data base synchronization, and backup/restore capabilities. These applications are often used to add resiliency to the IT environment and ensure that the IT organization has the ability to recover computing operations after catastrophic events. These bulk transfers using FTP over TCP are generally scheduled for off-peak hours to avoid conflict with other application traffic. However, by its very nature, the TCP windowing mechanism can make rather inefficient use of WAN link bandwidth, especially over high latency paths. With throughputs far lower than the bandwidth limit, job completion times can be very long, sometimes making it difficult to complete the job during off-peak time slots.

Recreation

User recreational activities include browsing the Internet, down loading music files, viewing Internet videos, peer-to-peer file sharing, and Internet/Intranet gaming. Recreational traffic has the potential to consume very large amounts of bandwidth and therefore requires the ability to classify application flows as recreational and apply QoS functionality to limit either aggregate bandwidth or per session bandwidth in order to protect business traffic from bandwidth starvation.

Malware

Many forms of malicious software attempt to use the network to infect as many end systems as possible. In addition, some types of viruses seek to attack the network infrastructure itself by corrupting routers or launching DDOS attacks. Therefore, Intrusion Prevention System (IPS) functionality in-line with the Intranet WAN may be required to protect WAN bandwidth by providing real-time mitigation of attacks and isolation of infected end systems.

4.0 WAN Bandwidth Optimization Techniques

There are two basic approaches to optimizing the use of WAN bandwidth. The first of these is to prioritize application traffic and to utilize QoS functionality to ensure that

mission critical and delay sensitive real-time applications receive the appropriate levels of service.

The second approach is to minimize consumption of WAN bandwidth by using a variety of data compression and data caching techniques to reduce the size of the data sets that actually traverse the WAN. The remainder of this section of the white paper will describe some of the specific techniques that can be used to reduce the amount of data sent over the WAN.

Static Data Compression

Static 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. Gzip is the primary compression algorithm used by HTTP V1.1.

LZ develops a codebook or dictionary as it processes the data stream and builds short codes corresponding to sequences of data. Repeated occurrences of the sequences of data are then replaced with the codes. The LZ codebook is optimized for each specific data stream and the decoding program extracts the codebook directly from the compressed data stream. LZ compression can often reduce text files by as much as 60-70%. However, for data with many possible data values LZ may prove to be quite ineffective because repeated sequences are fairly uncommon.

Differential Compression

Differencing algorithms are used to update files by sending only the changes that need to be made to convert an older version of the file to the current version. Differencing algorithms partition a file into two classes of variable length byte strings: those strings that appear in both the new and old versions and those that are unique to the new version being encoded. The latter strings comprise a *delta file,* which is the minimum set of changes that the receiver needs in order to build the updated version of the file.

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. In fact, a factor of ten or higher compression is not uncommon. As a result, differential compression can greatly reduce bandwidth requirements for functions such as software distribution, replication of distributed file systems, and file system backup and restore.

Real Time Dictionary Compression

The same basic LZ data compression algorithms discussed earlier can also be applied to individual blocks of data rather than entire files. Operating at the block level results in smaller dynamic dictionaries that can reside in memory rather than on disk. As a result,

the processing required for compression and decompression introduces only a small amount of delay, allowing the technique to be applied to real-time, streaming data.

Byte Caching

With byte caching the sender and the receiver maintain large disk-based caches of byte strings previous sent and received over the WAN link. As data is queued for the WAN, it is scanned for byte strings already in the cache. Any strings that result in *cache hits* are replaced with a short token that refers to its cache location, allowing the receiver to reconstruct the file from its copy of the cache. With byte caching, the data dictionary can span numerous TCP applications and information flows rather than being constrained to a single file or single application type.

Object Caching

Object caching store copies of remote application objects in a local cache server, 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. For example, in Web object caching, the 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 up to date.

If the local proxy does not contain a current version of the remote object, it must be fetched, cached, and then forwarded to the requester. Loading the remote object into the cache can potentially be facilitated by either data compression or byte caching.

5.0 Protocol Acceleration Techniques

A number of popular application protocols do not perform very well over enterprise wide area networks because their designs were optimized for other environments. For example, even though TCP was intended for wide area use, it was designed primarily to ensure reliability in lossy network and to maximize the fairness with which multiple TCP sessions share Internet bandwidth rather than to optimize per-session performance. Other protocols, such as NFS and CIFS, were designed primarily for a LAN environment, where the high bandwidth and low latency are compatible with highly granular client/server interactions.

Protocol acceleration refers to a class of techniques that improves application performance by circumventing the shortcomings of various communication protocols. Protocol acceleration is typically based on per-session packet processing by appliances at each end of the WAN link, as shown in Figure 1. The appliances at each end of the link act as a local proxy for the remote system by providing local termination of the session. Therefore, the end systems communicate with the appliances using the native protocol,

and the sessions are relayed between the appliances across the WAN using the accelerated version of the protocol or using a special protocol designed to address the WAN performance issues of the native protocol.

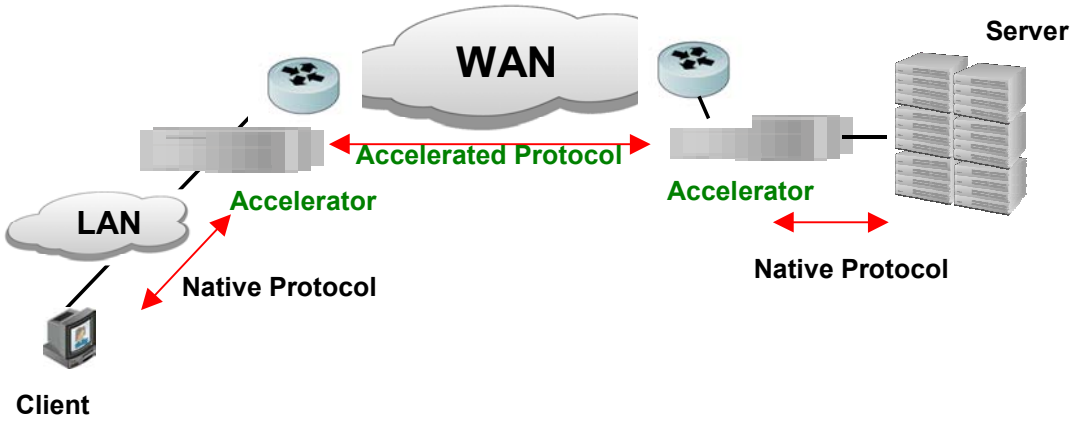

Figure 1: Protocol Acceleration Appliances

TCP Acceleration

TCP can be accelerated between appliances with a variety of techniques that increase a session's ability to more fully utilize link bandwidth. Some of the available techniques are dynamic scaling of the window size, packet aggregation, selective acknowledgement, and TCP Fast Start. Increasing the window size for large transfers allows more packets to be simultaneously in transit boosting bandwidth utilization. With packet aggregation, a number of smaller packets are aggregated into a single larger packet, reducing the overhead associated with numerous small packets. TCP selective acknowledgment (SACK) improves performance in the event that multiple packets are lost from one TCP window of data. With SACK, the receiver tells the sender which packets in the window were received, allowing the sender to retransmit only the missing data segments instead of all segments sent since the first lost packet. TCP slow start and congestion avoidance lower the data throughput drastically when loss is detected. TCP Fast Start remedies this by accelerating the growth of the TCP window size to quickly take advantage of link bandwidth.

CIFS and NFS Acceleration

As mentioned earlier, CIFS and NFS use numerous Remote Procedure Calls (RPCs) for each file sharing operation. NFS and CIFS suffer from poor performance over the WAN because each small data block (4KB) must be acknowledged before the next one is sent. This results in an inefficient ping-pong effect that amplifies the effect of WAN latency. CIFS and NFS file access can be greatly accelerated by using a WAFS transport protocol between the acceleration appliances. With the WAFS protocol, when a remote file is accessed, the entire file can be moved or pre-fetched from the remote server to the local appliance's cache. This technique eliminates numerous round trips over the

WAN. As a result, it can appear to the user that the file server is local rather than remote. If a file is being updated, CIF and NFS acceleration can use differential compression and block level compression to further increase WAN efficiency.

HTTP Acceleration

Web pages are often composed of many separate objects, each of which must be requested and retrieved sequentially. Typically a browser will wait for a requested object to be returned before requesting the next one. This results in the familiar ping-pong behavior that amplifies the effects of latency. HTTP can be accelerated by appliances that use pipelining to overlap fetches of Web objects rather than fetching them sequentially. In addition, the appliance can use object caching to maintain local storage of frequently accessed web objects. Web accesses can be further accelerated if the appliance continually updates objects in the cache instead of waiting for the object to be requested by a local browser before checking for updates.

Microsoft Exchange Acceleration

Most of the storage and bandwidth requirements of email programs, such as Microsoft Exchange, are due to the attachment of large files to mail messages. Downloading email attachments from remote Microsoft Exchange Servers is slow and wasteful of WAN bandwidth because the same attachment may be downloaded by a large number of email clients on the same remote site LAN. Microsoft Exchange acceleration can be accomplished with a local appliance that caches email attachments as they are downloaded. This means that all subsequent downloads of the same attachment can be satisfied from the local application server. If an attachment is edited locally and then returned to via the remote mail server, the appliances can use differential file compression to conserve WAN bandwidth.

6.0 Summary and Call to Action

This white paper described a number of factors that are making application delivery continually more complex. It is highly unlikely that the affect of these factors will lessen anytime soon. Because of that, IT organizations need to develop effective processes for ensuring acceptable application performance. This section of the white paper will identify some steps that IT organizations should take as part of creating those processes.

Establish a Policy for Acceptable Use

Companies need to establish a policy for the acceptable use of their IT resources. Undoubtedly, that policy needs to state that all forms of malicious traffic are unacceptable. It is highly recommended that the policy also state that any form of traffic that could get the company in legal trouble is also unacceptable. Examples of this include pornography and gambling. The policy must also deal with other forms of recreational traffic, such as Internet radio. One of the most compelling reasons for banning this traffic is that it has the potential to consume vast quantities of WAN

bandwidth. The need, however, to conserve bandwidth has to be evaluated in the context of the company's culture. For example, is allowing an employee to do online shopping while at work something that is in keeping with how the company wants to treat its employees? Given the complexity of these decisions, the creation of an acceptable use policy is not something that most IT organizations can do on its own. To create the policy, the IT organization will need the involvement of other organizations, such as the company's human resources organization.

Identify Applications

Successful application delivery requires that IT organizations are able to identify the applications running on the network and are also able to ensure the acceptable performance of the applications relevant to the business while controlling or eliminating applications that are not relevant. Referring back to section 2, the identification of the applications should include an understanding of whether or not the application is mission critical, the type of functionality provided and what type of demands the application makes of the network. For example, voice traffic is real-time and in most organizations would be deemed to be mission-critical. In contrast, Internet radio, which is also real-time, would not typically be deemed to be mission critical.

Focus on Mission-Critical Applications

As stated, there are typically only a handful of applications that are truly mission-critical. In order to be successful with application delivery, IT organizations absolutely must understand what those applications are and as outlined in section 3, must also understand what types of network situations cause these applications to perform badly. Having this understanding will position the IT organization to reduce the occurrence of those situations and hence ensure acceptable application performance.

Establish Application Service Level Agreements (SLAs)

Most IT organizations do not have SLAs for the performance of even their mission critical applications. Without an SLA for application performance it is difficult to say definitively whether or not an application is performing well. It is also difficult to justify an IT investment if the goal of that investment is to improve application performance.

IT organizations that are just getting started with crafting an SLA for the performance of their mission critical applications may want to avoid publicizing the SLA for a period of time. In particular, given the difficulty associated with application performance SLAs, IT organizations may want to have one in place for a long enough time that they feel comfortable that they can achieve the SLA before publicizing it.

Implement Quality of Service (QoS)

Given that there is a monthly recurring cost associated with the WAN and that this cost increases as the capacity of the WAN increases, few IT organizations can afford to

deploy a WAN where the capacity of every link is greater than the peak traffic demands. As a result, virtually all WANs will experience periods of congestion. As described in section 2, network congestion causes many applications to perform badly. QoS is necessary to ensure that mission critical, delay-sensitive applications receive the appropriate levels of service.

Evaluate WAN Bandwidth Optimization Techniques

Section 4 described a number of techniques that can be used to minimize the consumption of WAN bandwidth. IT organizations should evaluate the impact that these techniques have on the performance of their applications. While third party tests are helpful, wherever possible IT organizations should test the impact of these techniques in an environment that resembles the company's production environment as closely as possible.

Avoid a narrow focus

The majority of IT organizations deploy an application delivery solution to solve a particular problem. Once they have deployed the solution, however, many IT organizations then use the same solution to solve other problems. For example, an IT organization that deploys an application delivery solution to assist with its deployment of VoIP may later use that same solution to enable them to backup their databases over the network vs. by using tapes. As a result, when evaluating application delivery solutions IT organizations should choose a solution that solves the current problem, but which also has the capability to solve other problems that the IT organization is likely to encounter.