Realtime
publishers
"Leading the Conversation"

The Shortcut Guide™ To

Optimized WAN Application Delivery

sponsored by

Blue Coat®

Ed Tittel

Copyright Statement

Chapter 3: WAN Optimization Tools, Techniques, and Technologies

Any good tool requires its builders to cycle from circumstances (necessity), to conceptualization and creation (invention), before they can explore those developmental processes that kick-start innovation and growth. Software and hardware are no different in this regard. Even when tools are short-lived and quickly replaced or retired, they often serve as a basis or foundation for future developments. WAN optimization tools borrow tips, tricks, and techniques from many other disciplines that are not unique to WAN solutions. Even so, WAN optimization solutions usually combine multiple approaches and algorithms, some disjoint and independent, others interrelated and interdependent, into a single framework designed to enhance WAN performance.

Outmoded legacy applications and protocols also guide us to better designs, in part by serving as base references for inspiration and new implementations. Likewise, tried and true optimization techniques from areas as diverse as operating systems and databases also play a role in inspiring and informing WAN optimization tools and technologies. Although WAN optimization draws its ideas and techniques from many different computing disciplines, it uses them in its own interesting and specific ways to reduce the amount of data that must traverse WAN links, the frequency at which such traversals must occur, and the kinds of communications and protocols used to communicate across them.

If you perceive earlier network protocols as having grandfathered a newer generation of performance-enhancing network processes, protocols, and procedures you already have a good grasp on where WAN optimization is headed. You might also notice that steady, incremental improvements have also helped to optimize routing behaviors in an ever-increasing number of ways. This lets all of us understand that the motivation behind improving network performance remains the same, even when the modus operandi changes. From the broadest possible perspective, then, WAN optimization is just one class of tools available to designers and managers to help them make the most of their networks.

What Makes WAN Optimization Count?

Seamless connectivity across distant and widely dispersed offices is essential for any global organization, especially those enterprises seeking to attain a truly global reach. Optimal WAN connectivity is essential when it comes to delivering services and information to anyone, anywhere, at any time. Disaster recovery, workflow automation (run books and IT process automation), server consolidation and voice services all require a well-designed and -managed WAN infrastructure to become truly effective, and to ensure an acceptable return on the investments involved in acquiring such capabilities.

Indeed WAN optimization is particularly important for many reasons; here, we clarify some of the most important ones that lead enterprises and organizations into investing in such technology. In terms of cost, the expense of WAN links is a primary drag upon their initial adoption and consequently, on exploiting WAN connectivity to the fullest extent possible. WAN services may be delivered across many different types of media that include: leased-line, local loop, frame relay, ATM, DSL, Internet, and so forth. Assembling various WAN links and their communications to be consistent and coherent enterprise-wide remains a challenge, even when WAN optimization helps to make the most of what those links can carry.

Bandwidth allocation can also be crucial, when it comes to providing and supporting end-users in a cost-efficient, effective manner over the long-term. The same urgency applies to uninterrupted service and unimpeded communications anywhere they're needed, because they can make or break how business is conducted and carried out. Finally, support for widely-distributed users, services, and databases are integral to the maintenance, management, and monitoring in any network with broad geographic span, even if it's not truly global. That's why we discuss these various topics in more detail in the sections that follow.

Hierarchy as a Primitive Form of WAN Optimization

Organizations with complex network topologies spanning regional and territorial locations often utilize hub-and-spoke architectures for Wide Area Network (WAN) deployment. Each endpoint location represents a cog or *spoke* that links into a much larger *hub* represented by a headquarters data center. These larger networks might also include numerous smaller regional or territorial hubs—each one implementing its own territorial hub-and-spoke network—with any number and variety of high-bandwidth interconnections between them. This type of arrangement is depicted in Figure 3.1, but it's important to recognize that traffic volumes and priorities in this kind of scheme are built right into the architecture—namely, that higher volumes (and importance) attaches to traffic from the regional hubs to the central hub, and lower volumes (and importance) attaches to outbound traffic on the outermost links from regional hubs to branch offices.

Legacy WAN optimization schemes come from this hierarchical environment, where traffic optimization maps readily into link optimization. Such systems arrange large devices (or large collections of devices) at hub locations, with smaller devices at each spoke endpoint. Each spoke communicates with the hub through an explicitly-configured site-to-site *tunnel* through which compressed packets reach their final destinations. Endpoints, aka branch offices, seldom need to communicate with each other so they learn to live with the delays inherent in jumping from endpoint to regional hub to central hub and back out again. The hierarchical arrangement of network sites typical of a hub-and-spoke WAN topology appears in Figure 3.1.

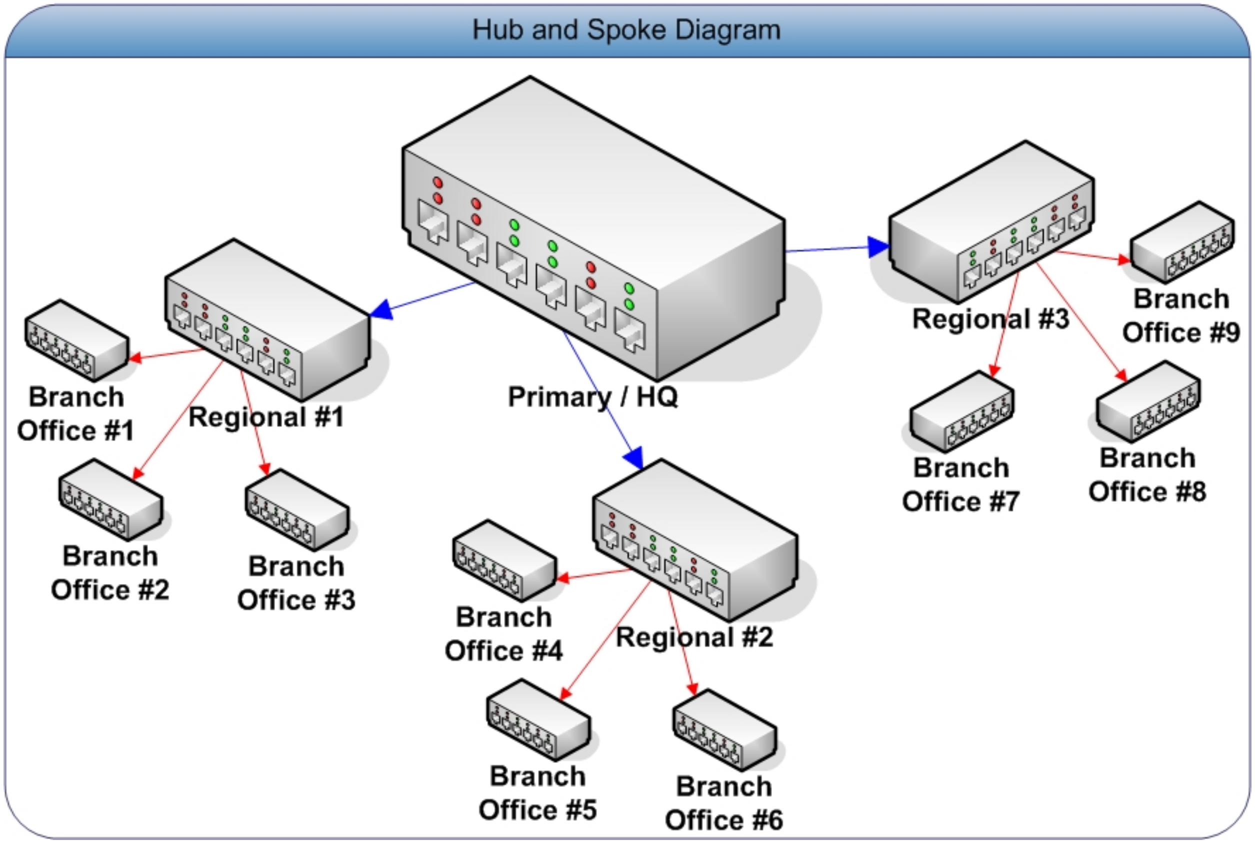


Figure 3.1: Blue links indicate “fat WAN pipes” between HQ and Regional hubs, red links “skinny WAN pipes” between Regional hubs and branch offices. Not shown: remote access links into all hubs!

But as the underpinnings of WAN technology have continued to evolve, a hierarchical, tunnel-based approach can be seen as an impairment rather than an improvement. Given the flexibility, scalability and performance available from more modern cloud architectures (which don’t need implicit or explicit hierarchy to function), the hub and spoke model can pose problems when changes in relationships, traffic patterns, or even work assignments overload WAN links at the periphery. Organizations and enterprises have found themselves scrapping hub-and-spoke architectures in favor of MPLS clouds, because these allow them faster access between arbitrary pairs of endpoints, and because additional carrying capacity can be laid on (or taken off) as changing traffic patterns and needs dictate.

Grabbing Text as a Primitive Form of Data Acquisition

Screen scraping is a term that refers to software techniques that open program windows, establish some kind of interactive session, initiate behavior, then read the results as they appear in the program's window. Thus, the notion of scraping the screen refers to grabbing the output text that appears thereupon to use as input to other programs and processes. Indeed, screen scraping represents a "lowest common denominator" form of software integration, and serves as the tool of last resort when grabbing data from any information source for which no other means of access or delivery is available. Where applications do offer any kind of more formal data access capabilities, typically through some kind of Application Programming Interface (API), there is no need for this type of practice. But where no direct APIs exist, screen scraping still prevails to this day. As painful as this process may sound, it's both a clever and brutal form of data acquisition. It is also pretty a pretty venerable practice in that it's been used for a long time, especially for legacy or older mainframe applications for which APIs aren't available and unlikely to be developed.

In essence, screen scraping is a method for manually processing and parsing character data to elicit its meaning and to supply record- or object-oriented data elements to other applications or services. Thus, screen scraping programs create links between modern platforms and legacy applications originally designed to work with inaccessible, inoperable or obsolete I/O devices and user interfaces. This extends their accessibility, and enhances the usability of associated logic and data, so that legacy programs and platforms can continue to serve some useful purpose.

Simplified computer interfaces that amount to text-based dumb terminals are often difficult to integrate with, or to interoperate with modern equipment. Elegant solutions require what may be non-existent luxuries: APIs, original documentation and source code, and legacy application programmers with experience on the target platform. Often the only practical solution is a screen scraper that acts as go-between for legacy and modern systems. A screen scraper may emulate command sequences or keystrokes to navigate the legacy user interface, process resulting display output, extract desired data, then pass it along to the modern system in whatever form and format it expects.

The concept of screen scraping is still utilized to harvest information in useful ways. Web scraping, a modern-age variant, generically describes any of several methods to extract content from Web sites to reformat or transform content into another context. Example scraper applications may scour retail sites—all coded in various languages and differently formatted—in search of books, cookware, and electronics categorized and indexed for online bargain hunters. Figure 3.2 shows a screen scraper at work, harvesting text from a Web browser and depositing same in a database.

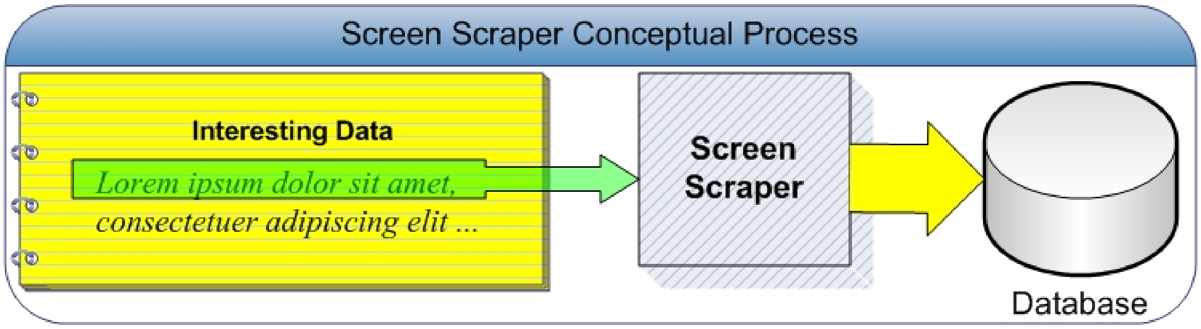

Screen Scraper Conceptual Process
Interesting Data
Lorem ipsum dolor sit amet,
consectetuer adipiscing elit ...
Screen Scraper
Database

Figure 3.2: A screen scraper operates a browser window just so it can harvest text on display there.

Screen scraping applications make excellent candidates for WAN optimization because they can fall prey to inefficiencies that WAN optimization tools address quite readily. First, they produce regular streams of character data that inevitably benefit from compression but also may benefit from dictionary and string caching capabilities. Second, screen scraping applications may utilize inefficient protocols, involve frequent communications, and be subject to “chatty” behavior. Properly repackaged through proxy agents, WAN optimization tools can help with all these shortcomings. But most important, the sheer doggedness of screen scraping as a technique for grabbing data when no other means is available shows us that clever programming techniques can also be applied when seeking to optimize WAN traffic, even if only at the level of brute force via compression or protocol streamlining.

Throttling Bandwidth Helps Manage Link Utilization

Treating symptoms rather than causes is a common but all too often unsuccessful solution for many problems, including bandwidth consumption. Results may appear positive at first, but these effects are generally short-lived, unreliable or unpredictable, and do not scale well. Bandwidth throttling reflects this approach by seeking to treat the symptoms (slow transfers or excessive response times) rather than the causes (low bandwidth availability, cyclical processing peaks, unauthorized applications, inappropriate resource links, and so forth). Nevertheless, this technique has yet to fall into disuse or total disrepute.

Bandwidth throttling ensures that bandwidth-intensive devices such as routers or gateways limit the quantities of data transmitted and received over some specific period of time. Bandwidth throttling limits network congestion and reduces server instability resulting from network saturation. For ISPs, bandwidth throttling restricts user speeds across certain applications or during peak usage periods. But without understanding the causes of traffic spikes that require links to be throttled, it’s always just a matter of time before the load comes back again, ready to be throttled again.

Data-bearing servers operate on a simple principle of supply and demand: clients make requests, and servers respond to them. But Internet-facing servers that service client requests are especially prone to overload during peak operating hours and under heavy, intense network loads. Such peak load periods create data congestion or *bottlenecking* across the connection that can cause server instability and eventual system failure, resulting in downtime. Bandwidth throttling is used as a preventive method to control the server's response level to any surges in client requests throughout peak hours of the day.

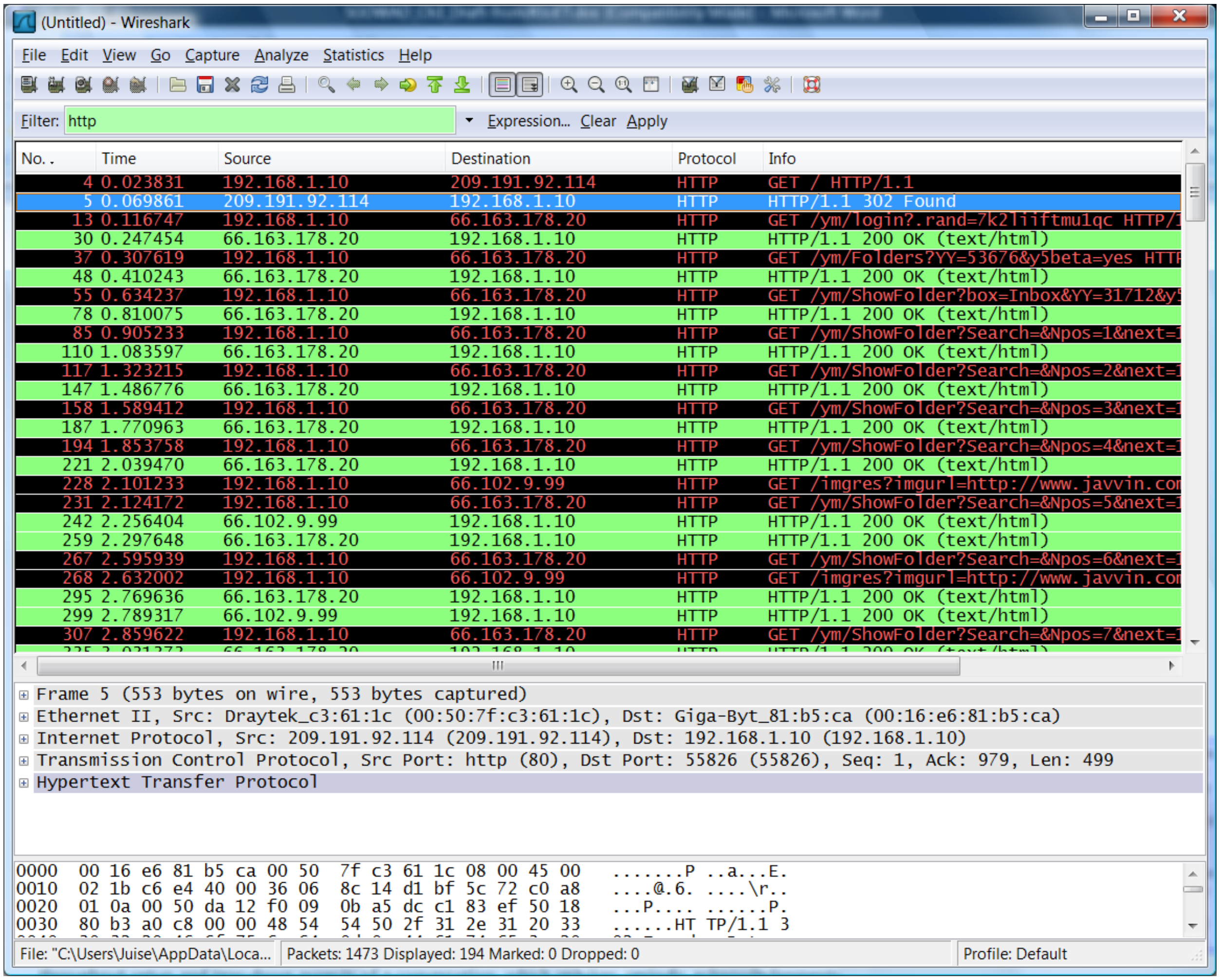
In February of 2008, members of the Federal Communications Commission announced they might consider establishing regulations to discourage Internet providers from selectively throttling bandwidth from sites and services that would otherwise consume large amounts. In late 2007, Comcast actively interfered with some of its high-speed Internet subscribers using file-sharing clients and protocols by throttling such connections during peak hours (and only for uploads). This sparked a controversy that continues to this day.

Organizations can (and should) use bandwidth throttling or firewall filters to limit or block traffic that explicitly violates Acceptable Use Policy. But otherwise, bandwidth-throttling is best applied in the form of Class of Service or Quality of Service (CoS/QoS) markers applied to various types or specific instances of network traffic. CoS and QoS represent classification schemes for network traffic that give priority to time-sensitive and mission-critical traffic rather than by limiting a specific type of traffic explicitly. Many experts recommend that unauthorized or unwanted protocols be throttled to extremely low levels of bandwidth (under 10 Kbps) rather than blocked completely, so as to give network administrators an opportunity to ferret out and deal with users or programs involved. Thus, for example, by limiting bandwidth available to peer-to-peer protocols such as BitTorrent (used for video and other personal media downloads) or FastTrack (the Kazaa protocol) to only 5K or 10K bits per second, administrators may have time to identify the workstations or servers acting as endpoints for related peer-to-peer activities, and identify the individuals involved in their use. They can then counsel or discipline users as per prevailing acceptable use policies (AUP).

Fixing WAN-Unfriendly LAN Protocols

As we discussed initially in Chapter 1, many existing protocols that support vital internal and external business functions operate with limited scope and scalability. These are almost always LAN-based legacy protocols that can impose a serious drag upon wide-scale WAN. For example, the Common Internet Files System, aka CIFS, grows exponentially unwieldy in transit across WAN linkages. Likewise, real-time streaming and voice protocols often introduce needs for *traffic shaping*. In this practice, controls are applied to network traffic so as to optimize performance, meet performance guarantees, or increase usable bandwidth, usually by delaying packets that may be categorized as relatively delay insensitive or that meet certain classification criteria that mark such traffic as low priority. Traffic shaping is often employed to provide users with satisfactory voice and video services, while still enabling "networking as usual" to proceed for other protocols.

In the same vein, encryption and security protocol acceleration tools become resource-intensive but utterly necessary burdens, especially when sensitive traffic must traverse Internet links. Even the most widely used protocol on the Internet—namely, HTTP—may be described as both chatty (involving frequent communications) and bursty (involving numerous periods during which tens to hundreds of resource requests may be in flight on the network at any given moment). The protocol trace shown in Figure 3.3 indicates that the display of a single Web page, involves back-and-forth exchange of information about a great many elements over a short period of time (12 showing on the sample trace, with more out of sight below).


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Help
Expression... Clear Apply

Filter: http

No. .	Time	Source	Destination	Protocol	Info
4	0.023831	192.168.1.10	209.191.92.114	HTTP	GET / HTTP/1.1
5	0.069861	209.191.92.114	192.168.1.10	HTTP	HTTP/1.1 302 Found
13	0.116747	192.168.1.10	66.163.178.20	HTTP	GET /ym/login?.rand=7k2liiftmu1qc HTTP/1
30	0.247454	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
37	0.307619	192.168.1.10	66.163.178.20	HTTP	GET /ym/Folders?YY=53676&y5beta=yes HTTP
48	0.410243	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
55	0.634237	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?box=Inbox&YY=31712&y5
78	0.810075	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
85	0.905233	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=1&next=1
110	1.083597	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
117	1.323215	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=2&next=1
147	1.486776	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
158	1.589412	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=3&next=1
187	1.770963	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
194	1.853758	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=4&next=1
221	2.039470	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
228	2.101233	192.168.1.10	66.102.9.99	HTTP	GET /imgres?imgurl=http://www.javvin.com
231	2.124172	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=5&next=1
242	2.256404	66.102.9.99	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
259	2.297648	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
267	2.595939	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=6&next=1
268	2.632002	192.168.1.10	66.102.9.99	HTTP	GET /imgres?imgurl=http://www.javvin.com
295	2.769636	66.163.178.20	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
299	2.789317	66.102.9.99	192.168.1.10	HTTP	HTTP/1.1 200 OK (text/html)
307	2.859622	192.168.1.10	66.163.178.20	HTTP	GET /ym/ShowFolder?Search=&Npos=7&next=1

```
⊞ Frame 5 (553 bytes on wire, 553 bytes captured)
⊞ Ethernet II, Src: Draytek_c3:61:1c (00:50:7f:c3:61:1c), Dst: Giga-Byt_81:b5:ca (00:16:e6:81:b5:ca)
⊞ Internet Protocol, Src: 209.191.92.114 (209.191.92.114), Dst: 192.168.1.10 (192.168.1.10)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: 55826 (55826), Seq: 1, Ack: 979, Len: 499
⊞ Hypertext Transfer Protocol
```

```
0000  00 16 e6 81 b5 ca 00 50  7f c3 61 1c 08 00 45 00   .......P ..a...E.
0010  02 1b c6 e4 40 00 36 06  8c 14 d1 bf 5c 72 c0 a8   ....@.6. ....\r..
0020  01 0a 00 50 da 12 f0 09  0b a5 dc c1 83 ef 50 18   ...P.... ......P.
0030  80 b3 a0 c8 00 00 48 54  54 50 2f 31 2e 31 20 33   ......HT TP/1.1 3
```

File: "C:\Users\Juise\AppData\Loca... | Packets: 1473 Displayed: 194 Marked: 0 Dropped: 0 | Profile: Default

Figure 3.3: A single Web page fetch can spawn tens to hundreds of HTTP "Get" requests and associated data-bearing replies.

Fixing these "broken" aspects of the network environment becomes a traffic engineering proposition that takes into account not just the applications themselves but application programming in general. Knowing how an application operates, its protocol formats and parameters, and observable run-time behaviors is crucial to understanding how it fits with other applications, services, and protocols on the network. It's not just a patchwork proposition that involves mending individual parts, but instead requires accommodating best practices for efficient WAN communications: send and receive infrequently, in bulk, and in the form of complete transactions whenever possible.

The TCP format was originally designed and engineered to operate reliably over unreliable transmission media irrespective of transmission rates, inherent delays, protocol corruption, data duplication, and segment reordering. Because of this, TCP is indeed a robust and reliable mechanism for delivery of applications, services, and protocols. But this design strength also exposes inherent weakness in TCP delivery when deployed across modern, higher-speed media that completely exceed the conditions under which TCP was originally intended to be used.

Re-engineering these and other “broken” network protocols occurs in WAN optimization solutions, usually through some form of proxy. Such a proxy permits unfettered protocol behavior so that protocols can behave normally and unhindered on the LAN. But the same proxy also translates and typically repackages LAN-oriented transmissions to reduce or eliminate “chattiness” across WAN links, while also batching up individual transmissions to limit the number and maximize the payloads for such WAN transmissions as do occur. This approach maximizes use of available bandwidth when transferring request and reply traffic across WAN links.

IP blindly sends packets without checking on their arrival; TCP maintains ongoing end-to-end connections throughout setup and tear-down phases, and even requires periodic acknowledgements for receipt of data. Unacknowledged data triggers an exponential back-off algorithm that times out and retries transmissions until they’re received and acknowledged, or times out to signal connection failure. Sliding TCP window sizes—these denote the number of packets that can be sent before receipt of an acknowledgement is required—directly influences performance where larger values equal greater throughput (but also, much longer potential delays). TCP employs a well-defined “slow start” algorithm that initiates communications with a small window size, then scales TCP window sizes to optimal proportions as connections are established and maintained while they remain active. Each of these and other such procedures of the TCP/IP stack introduce network delay addressed in WAN optimization solutions through connection optimization techniques and aggressive windowing methods.

📖 For an outstanding discussion on TCP window size, the slow start algorithm, and other TCP congestion management techniques, please consult Charles Kozierok’s excellent book *The TCP/IP Guide*. This book is available in its entirely online; the section on TCP Reliability and Flow Control Features and Protocol Modifications includes detailed discussion of TCP window management, window size adjustment, congestion handling, and congestion avoidance mechanisms.

Use of Compression to Reduce Bandwidth

As a result of their ability to reduce traffic volume, various performance-improving compression schemes have been an integral part of Internetwork communications since way back when X.25 and Bulletin Board Systems (BBSs) used the ZMODEM file transfer protocol introduced in 1986. Over twenty years later, compression regularly appears in more modern network-oriented protocols like Secure Shell (SSH) and byte-compressing data stream caching in contemporary WAN optimization products.

In telecommunication terms, *bandwidth compression* means: a reduction of the bandwidth needed to transmit a given amount of data in a given time; or a reduction in the time needed to transmit a given amount of data within a given amount of available bandwidth. This implies a reduction in normal bandwidth (or time) for information-bearing signals without reducing the information content, thanks to proper use of data compression techniques. These are well and good in the WAN environment, but can be ineffective without access to accelerated compression hardware to achieve maximum compression and decompression with minimum time delays (though software is fast nowadays, hardware is usually several orders of magnitude faster—an essential characteristic, given modern WAN link speeds). WAN optimization devices generally include symbol and object dictionaries to reduce data volumes even more than compression can provide alone, and are discussed later in this chapter.

Recurring redundancy, which describes the ways in which patterns and elements tend to repeat in regular traffic streams between pairs of senders and receivers, remains the crucial reason why compression schemes still thrive. From the right analytical perspective, much of the data transiting networks includes unnecessary repetition, which wastes bits and the time and bandwidth necessary for their conveyance. Compressing data by all possible means helps restore balance to the networking order and is just one of several counters against unwanted network delay.

Various types of compression techniques are eminently suitable for network media, but all of them strive to reduce bandwidth consumed during WAN traversal. Header and payload compression techniques utilize pattern-matching algorithms to identify short, frequently recurring byte patterns on the network that are replaced by shorter segments of code to reduce final transmitted sizes. Simplified algorithms identify repeat byte patterns within individual packets where sophisticated forms of compression may analyze patterns across multiple packets and traffic flows.

Any gains that compression strategies provide must vary according to the mix and makeup in WAN traffic. Compressed archives of data (such as ZIP or tar files) cannot be reduced much further using network compression schemes, but applying compression across various flows of traffic can still enhance effective WAN bandwidth. Voice protocols significantly benefit from UDP header compression in conjunction with other techniques such as packet coalescing, described in the following section.

Redundant Overlays Bring Files Closer to Users

Redundancy isn't always bad for network performance or throughput. There are many applications and instances where multiplicity can enhance performance. In fact, WAN environments benefit greatly from using redundant overlays of file services that bring files and data closer to their end-users.

As the DNS system has shown us, a network can operate effectively in consolidated and decentralized ways at the same time. A single, uniform body of information can be consolidated from many sources and distributed throughout a completely decentralized system for delivery anywhere in the networked world. Consolidating aggregate data in a single location only creates benefits for those users likely to be in close network proximity to the data center, but can pose accessibility issues for other users more than one network hop away, especially those who must use narrow or expensive WAN links to connect to that data.

Resorting to a central authority or source of information for a globally-dispersed company can have predictable negative issues for remote and roaming users, so it's better to replicate information in places where its users can access it quickly, no matter where they might be. DNS databases, with their masters and slaves, and authoritative and non-authoritative versions, along with careful use of caching of recent activity, create a model for widespread, dispersed use of distributed information that works well enough to keep the global Internet humming along. In similar fashion, redundant overlays seek to keep the files that users are most likely to access no more than one or two WAN hops away from their machines, no matter where they might be at any given moment in time.

Today, many companies toil with the challenge of server consolidation and proliferation. Numerous in-house servers service a variety of application services and network protocols, and despite their overwhelming ubiquity they aren't always in the right place to serve at the right time. Many companies opt instead to roll out several key servers in strategically-located installations throughout their geographical and territorial boundaries to better accommodate "away" teams and users. This approach permits a group of carefully synchronized servers that handle multiple basic business needs to deliver comparable accessibility, bandwidth, and security to anyone anywhere. Replication across multiple physical servers makes this approach possible, while virtualization so that individual services run in separate virtual code spaces makes the process more practical and maintenance and monitoring more workable.

Network assets and end-users are often dispersed across branch offices, customer sites, and ISPs that span multiple regions. A well-designed server consolidation strategy necessarily centralizes the infrastructure and reduces server count to save on costs and improve management. Unfortunately, this also effectively places the burden of ensuring seamless connectivity between remote locations directly onto WAN links, and fails to deliver the goods whenever such links fail. This means there must be some mechanism or mechanisms in place to pick up traffic that occurs as the number of remotely-connected users increases.

During opening hours, many businesses endure a surge of initial network traffic that consists largely of multiple users logging in simultaneously to one or more servers. Authentication and DNS directory services access is commonly problematic during this pre-game warm-up routine where everyone shows up to log-in at the same time, so there needs to be some way to optimize and prioritize traffic so wait times at login prompts are minimal. A consolidated WAN optimization solution helps obtain the best use of network architecture because it confers an ability to make the most of the bandwidth a WAN link can carry, while also optimizing priority traffic so that when congestion occurs important data gets through anyway. Thus, the morning user, coffee in one hand, keyboard in the other, might have to wait a minute (literally) to download e-mail, but he or she will still get a quick response when they hit return after supplying an account name and password.

State-of-the-Art Acceleration Techniques

Technology has become another greatly anticipated (though artificial) evolution for mankind ever since the big initial discoveries of man-made fire and the wheel. Technology continues to evolve to suit a greater variety of purposes or deliver an ever-increasing range of functionality. Technology also continually adapts and occasionally demands redesigns to remain viable in an ever-changing environment, until it eventually reaches retirement age and enlistment into the annals of historical or disused technologies.

Technology replacement often comes swiftly, where replacements often provide vast improvements over original designs. Any-to-any network clouds, which utilize MPLS or ATM infrastructures to create and manage links between arbitrary pairs or collections of sites using high-speed WAN links, have an increasing presence in the network landscape, but carry along the inherent issue of too many tunnels because WAN optimization devices usually work in pairs rather than in groups. Even with completely error-free, fully-operational WAN links there are performance-dampening provisioning technologies and practices causing packet-loss at the network layer. Combined with high-latency, timed retransmissions, and congestion-avoidance behaviors native to TCP, this problem can cause application performance to suffer perceptibly even when bandwidth is available and affordable. It’s not enough to have a WAN to use, it’s also important to understand how to make best use of that resource as well.

One problem for network cloud operation comes from a preponderance of encapsulated tunnel traffic between pairs of peers. The notion is that traffic flows from two sets of networks through an established tunnel that moves traffic through the network medium between endpoints to that tunnel. WAN optimization must be sensitive to this type of communication flow, and make intelligent use of tunnel and receiver connection setup and maintenance to use no more bandwidth or connections than are absolutely necessary when communications move through a cloud.

Indeed, many organizations now utilize any-to-any network clouds that replace the hub-and-spoke paradigm mentioned earlier in this chapter, where these clouds are often based and built upon Multi-Protocol Label Switching (MPLS). This next-generation forwarding and switching architecture realizes WAN “cloud” deployment complete with WAN optimization strategies and important advantages for advanced services and traffic engineering.

Disaster recovery sites require redundant high-speed WAN links for timely and critical data backups to be both accurate and effective at all times. Workflow automation accounts for internal business processes, internal business personnel, and the entire evolution of the business ecosystem including suppliers, partners and customers. Neither scenario is satisfied nor well-served by improperly or poorly-managed WAN links. What happens when crucial customer orders need priority reception? You must ensure delivery of critical business processes for the ordering system across the WAN link, which may involve WAN-applicable Quality of Service (QoS) policies. Orders expedited to manufacturing plant enterprise resource planning (ERP) systems may even require dedicated leased-line linkages to HQ offices. Here again, we find multiple opportunities for WAN acceleration to move information faster and more efficiently using heavily-shared, heavily-utilized network WAN links.

Opening Closed Communications Delivers Desirable Results

Our technological approach and intellectual analysis of the greater framework involving networking applications, protocols and technologies expands both how we conceive the network (as a whole) and how we deploy its various components to satisfy operational needs. We no longer see communications as “I need to send this message or file from point A to point B”, but instead take a deeper analytical approach and explore the parameters that define that message or file, how it travels across a number of links, and what operational inconveniences it imposes. We see the protocol fields and values that influence its routing behavior and the routing characteristics that define its path and progress through the network.

Using the appropriate viewfinders, we can even peer into and visualize our network as a much larger landscape comprising a universe unto itself. We can open, inspect and accelerate Secure Socket Layer (SSL) applications, which is absolutely vital for WAN optimization. Encrypted network data streams are invisible unless they can be opened and handled somehow, which is why WAN optimization devices are often furnished with the necessary encryption keys and certificates to handle those tasks. This enables WAN optimization to peer inside encrypted traffic to analyze data redundancy and lets it enforce traffic policies it might not otherwise be able to invoke. This view inside encrypted streams also makes it possible to analyze and define traffic patterns to apply classic compression techniques, and to use shared symbol and object dictionaries to further reduce the volume of data in motion. This capability is depicted in Figure 3.4.

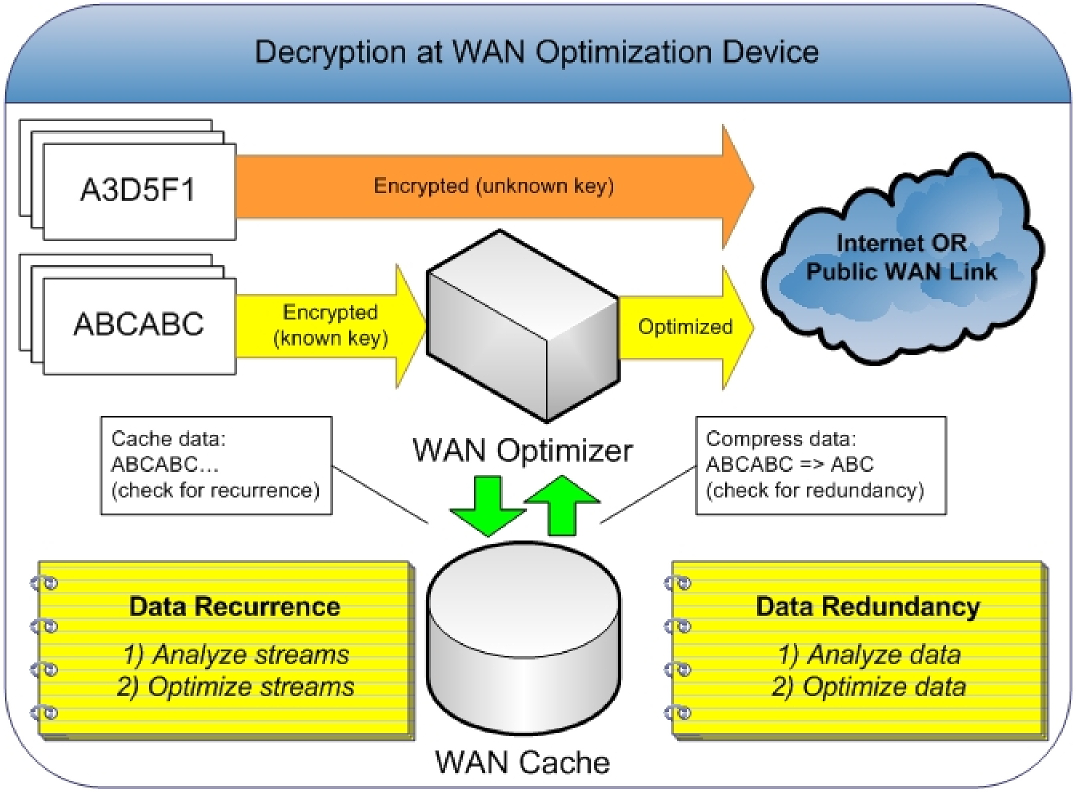


Figure 3.4: Given the ability to decrypt encrypted data streams, WAN optimization devices can enforce policy, impose throttling, and even apply various compression and dictionary schemes.

Advanced Acceleration Techniques Speed Streaming Voice and Video

Latency mitigation strategies reduce and shorten delivery times between sender and receiver, which effect increases proportionally constant to the distance travelled. In other words, distance introduces delay; longer distances introduce longer delays. Latency is also increased in transit by queuing and processing through intermediary network appliances and routing devices. These incremental delays levy appreciable impacts on WAN performance, where severe latency incurred from overly chatty TCP conversations and payload-bearing bulk transfers can potentially sever throughput.

Local caching at endpoints is an excellent strategy for delivering information efficiently at reduced data transmission levels. By keeping recent copies of large or repeated requests and transfers nearby, less redundancy and bandwidth need be consumed when passing such data to its consumers. However, caching does require a more analytical and focused perspective on ongoing network communications along with access to the right on-site caching mechanisms to be truly effective.

Flexible Bandwidth Controls Provide Multiple Benefits

Well-defined policies should drive flexible bandwidth controls to throttle junk traffic and ensure better response time for important traffic. High-level application content and user information weighs heavily upon formulating such policies, which are best orchestrated by involving executive staff, key stakeholders, management, and representatives from the user community. It’s also essential to test the impact of resulting policy schemes on the user experience, and to keep ongoing tabs on that experience to adjust policy settings over time to reflect changing needs and priorities.

Acceptable use policy also plays a surprisingly important role in maximizing the value and use of WAN links. It’s well known that reducing network noise enhances signal clarity; it’s less clear to many (especially undereducated users) that wasting bandwidth on non-work related activities really can place a serious and unwelcome drain on the bottom line. Eliminating unnecessary and undesirable traffic can be a major factor when imposing network traffic management policies. Such controls must be well-considered, carefully deployed, and flexible enough to accommodate the many different types of (authorized) traffic typical on modern enterprise networks.

Traffic by Type and Source

In some cases, traffic policing or WAN optimization may fail to preserve original client-server address designations and protocol formats, or to keep such data easily accessible. That’s because of the many layers of embedding and encapsulation that can occur at various layers of the protocol stack. Applications, WAN optimization tools, and traffic policies that aren’t careful to observe or maintain sufficient original context are in for worlds of trouble if they block or downgrade priority on traffic that really needs to get through.

Monitoring solutions must be sensitive to both original and encapsulated formats so as to accurately and effectively report end-to-end performance and traffic conditions. Many WAN optimization solutions tunnel protocol traffic that's necessary for this monitoring process to work properly and thereby obscure network conditions. A monitoring system can lose visibility into individual application flows as they disappear into optimized WAN ports or tunnels. That's why it's important to ask about such behaviors, and to tell WAN optimization vendors what kinds of management and flow related data they should capture and relay for monitoring and management purposes.

Traffic by Time of Day

Peak hours for various procedures, processes, sites, and users generally occur at regular, predictable intervals on the networks they use. Peak business hours can be observed, monitored and predicted over a given timeline and then used to construct a time-sensitive traffic management strategy. Offline hours then become peak activity windows for off-duty processes, such as daily accounting or inventorying systems, and network-based backup. Time of day is crucial for monitoring operational behaviors on the network and controlling resource availability, so also are periodic peaks that include end-of month, end-of-quarter, and end-of-year cycles, as well as occasional on-demand cycles (audits or emergencies, for example).

Application Traffic

An inordinate number of applications and application protocols exist that can be controlled and monitored consistently and cohesively Each obtains its own priority assignment, poses its own unique value in the network management equation. Not all applications are created equally, though many are designed equally badly (or are comparatively worse) when it comes to WAN deployment. The abilities that WAN optimization solutions confer to tame these sometimes savage beasts remain among their most potent value propositions.

Some vendors offer Web acceleration appliances that optimize only certain types of traffic by off-loading certain servers. Other products optimize all TCP traffic equally regardless of differences in their application-layer behaviors. A complete and comprehensive WAN optimization solution must be able to selectively prioritize traffic, especially in situations where WAN links are heavily-utilized or operating at (or above) their rated capacity.

Traffic by User or Group Identity

User and group identity for sender and receiver pairs is another parameter that may be factored into WAN optimization. Authentication is critical for WAN optimization because it enables acceleration or prioritization of traffic based on individual and group membership. This means that user identity and/or group membership may be used as a criterion for allowing or disallowing traffic, or for prioritizing some traffic, and not prioritizing others. For example, email bound to or from the CEO goes ahead of the line but the mailroom clerk has to wait for "slack" capacity for his messages to be handled.

User identity tracking also facilitates better end-to-end network visibility. It can allow network engineers and planners to streamline security and prioritize delivery of certain traffic from certain sources. Identity may be used to block or allow certain types of traffic, or to apply varying levels of priority to the same kinds of traffic (CEO and customer support email goes ahead of all other email, for example). In the same vein, a salesperson located in some remote branch office may be granted higher priority than a marketing staff employee when accessing the company's centralized CRM application, because of the perceived difference in importance for such access (servicing an existing customer in the former case, prospecting for new customers or expanding on an existing relationship in the latter case).

More About Compression Dictionaries and Local Caching

A compression dictionary is any location where an algorithm stores its data sequences, predictions, shortened substitutions and other process-related data. Each dictionary has an associated size governing the amount of data stored and retrieved for compression and decompression.

Compression algorithms shrink data transfer sizes without altering critical protocol payloads—a form of lossless compression. Many algorithms spot repeated sequences of data and store them for later look-up and retrieval. Some algorithms learn the series order of predictable data sequences rather than actual data content, and then predict subsequent content based on preceding patterns. Where such predictions prove correct, indicators of success are transmitted (as opposed to live, repeat data) to a compression partner on the receiving end. This partner notes the positive indicator and restores original data in place of the signifying notification.

Maximize Rapid Access to Frequently Used Files, Resources, and Data

In the realm of servers and workstations, disk defragmentation seeks to boost operational performance by resituating frequently and recently used files, resources, and information. A process of physical reorganization and analysis of access patterns helps determine what data and which files should fit together closely and contiguously for fast access.

In much the same way, frequently and recently-accessed network files, data, and resources should take priority over less frequently referenced elements. Some situations and scenarios call for decentralized servers to house consolidated information and resources as described earlier in this chapter when we discussed DNS decentralization. In any case, the notion is that by watching what kinds of access have repeated in the past, we can predict what kinds will repeat in the future, and use that information to populate local caches that can cut down on the need for WAN access.

Use Software Acceleration for Roaming Users on the Go

Endpoint client applications for WAN optimization are necessary in addition to the authentication-driven, identity-based traffic prioritization and tracking we mentioned earlier in this chapter Such applications open the boundaries of an enterprise and help make an optimization solution and strategy extend truly end-to-end, rather than remaining oriented primarily at specific tunnels between optimization devices (as is the case for most implementations). Furthermore, client-based software acceleration applications allow all kinds of policy-based controls and priorities to be applied, even when a client is operating independently in a Starbucks in Des Moines, far from her branch office in Dubuque 50 miles away.

Software acceleration clients benefit from built-in “situational awareness”—that is to say, from the client’s perspective the WAN optimization solution is aware of its location (branch office, remote VPN link, HQ) and can automatically apply an appropriate optimization strategy to match. Client software can approximate its proximity to the source server to determine its best possible routes and security options as well.

Caching

Caching is an excellent strategy in any aspect of computing. Router hardware caches MAC address tables and maintains lists of IP assignments; application proxies cache application layer data to conserve bandwidth against repeat requests; and WAN optimization technologies cache sequences of traffic data to avoid duplicate replay of protocol patterns on the network. And the process can be entirely application-independent for general-purpose usage. Single-purpose caches work only with specific applications or repeat requests for the same resource irrespective of all other network traffic (Web-only, email-only, backup-only, ERP and so forth). WAN optimization devices have a global view of all traffic that passes over the links they manage, so their caches can handle data for all the applications whose traffic traverses those links (making them a “links-only” rather than “application-only” type of cache).

Data reduction is an efficient means for WAN application and bandwidth optimization. The trick here is to avoid sending as much data as possible, or at least, never to send the same data more than once. Acceleration appliances examine data in real-time prior to its transmission across the WAN, and store objects and items locally. Any duplicate detected triggers the appropriate appliance to resend data locally instead of moving that same data (unnecessarily) across a WAN link.

Wherever data is stored by intermediary devices, it should also be handled in a secure, policy-driven manner. Caching copies of repeatedly issued data across a network is a great strategy for network performance but a terrible hindrance for application security across the data path. Any information obtained from this cache must also be secured so that it’s both accurate and timely upon delivery to the requesting source, but also safe and secure from unwarranted inspection or alteration by unauthorized third parties.

Ideally, a cache should also be free of deployment constraint. Transparency plays a crucial role in the peaceful coexistence of intermediary device and end-user, so having to expose caching servers and services through end-user configurations can be a labor-intensive hands-on process. Zero-configuration is the objective in many of today’s application platforms, and this area is no exception. Any and all network optimizations should be readily accessible and completely transparent to the end-user.

Coalescing Common Data Elements

Smaller recurring packets have repeatable header data patterns that consume substantial amounts of bandwidth, which grows comparatively exponential in relation to the payloads. Packet coalescing merges multiple packets into one, provided they traverse the same link to reach the same endpoints.

In combination with header compression, this applies single header across multiple packets to decrease operational overhead and achieve bandwidth requirements. Web, voice and interactive multimedia applications all benefit greatly from packet coalescence.

Bandwidth Control

Bandwidth control and bandwidth management are two ways of saying the same thing. It is the process of measuring and controlling packet-based network communications to avoid overusing capacity, which results in network congestion and poor performance. The channel capacity of partitioned, multiple-user Internetwork links is administratively limited. Once this threshold is reached, performance degrades in a highly noticeable way—network congestion.

Controlling and managing network traffic reduces capacity use to maintain smooth, continuous service between endpoints. The art and science of controlling and managing traffic is a deeply-faceted practice of its own, with myriad solutions at virtually every layer of the network protocol stack. ISPs typically retain control over queue management and QoS to subscribers, window shaping promotes traffic flow reduction in high-end enterprise products and other such solutions increase usability of network capacity and resources.

The majority of WAN protocols utilized today include Integrated Services Digital Network (ISDN), frame relay, Multi-Protocol Label Switching (MPLS), Asynchronous Transfer Mode (ATM), and Point-to-Point Protocol (PPP) over Synchronous Optical Network (SONET). Harmonizing and orchestrating optimal performance among this heterogeny requires handling a series of deeply complex tasks.

WAN Technologies Summarized

The following material contains introductory subject matter on the topic of current and existing WAN environments and technologies. Topics include point-to-point links, circuit and packet switching methodologies, virtual circuits, dial-up services, and WAN endpoint devices.

WAN data communications encompass broad geographically dispersed areas facilitated by transmission services of common carriers, as depicted in Figure 3.5. WAN technologies generally function at layers 1 through 3 of the OSI TCP/IP reference model, which is why most network engineers also function at this level. However, WAN appliances can also be "application aware" in that they are capable of analyzing application protocol streams and manipulating their behavior according to policy.

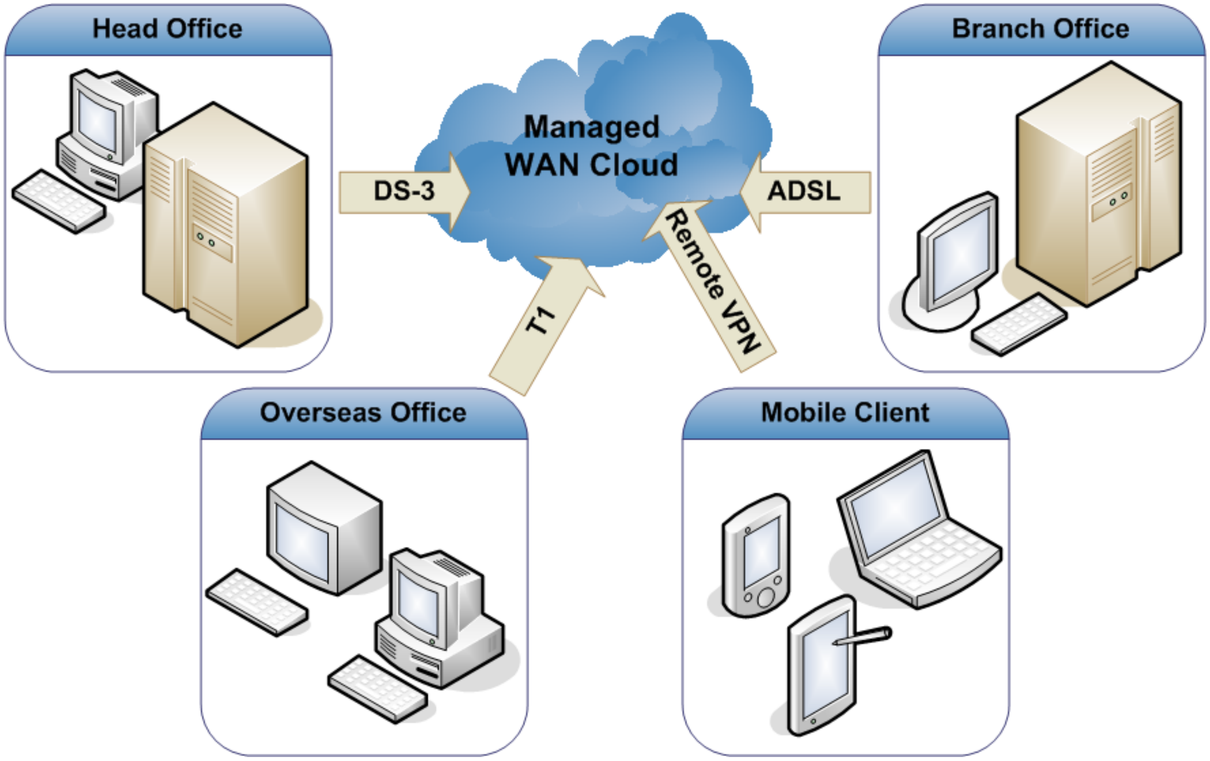


Figure 3.5: A key benefit of the "Managed WAN Cloud" is its ability to accommodate different kinds of WAN links for ingress and egress.

Point-to-Point Links

An established, individual communications path from subscriber to provider is referred to as a *point-to-point* link. In this arrangement, a carrier network (such as a local telephone company) provides a direct connection via leased lines (that may include copper wiring and other necessary hardware such as CSU/DSU units) to the customer's premises. Accordingly, both sets of links will generally use the same service provider network arrangements.

Circuits are normally priced according to bandwidth requirements and the distance between the two connection points. Point-to-point links are typically priced higher than Frame Relay links but also provide permanently established, exclusive connectivity between provider and subscriber regardless of the extent to which allocated bandwidth may be utilized. Another common term for such a link is *leased line* (which refers to the ongoing reservation of the connection between the two endpoints).

Circuit Switching

Using *circuit-switching* communications, data paths are formed as needed and terminated when such use ceases. This setup operates much like a typical telephone network in that "conversations" are arbitrarily created and terminated, existing only for the duration of the "call" (which is actually an active data connection between at least two parties).

ISDN is a primary example of this kind of technology: a switched circuit is initiated whenever a router possesses data for a remote site, which essentially places a direct-dial call into the remote site's circuit. Once the two parties are authenticated and connected, they begin the transfer of data from source to destination. Upon completion, the call terminates.

Packet Switching

WAN *packet-switching* technology uses a shared carrier infrastructure unlike the private, one-on-one pairings used in a circuit-switched network arrangement. This scenario enables the carrier to make more efficient use of its infrastructure, often resulting in better subscriber costs for similar levels of service. In a packet-switched environment, a shared WAN medium is distributed and utilized among a broad subscriber base that creates virtual connections between sites for packet delivery.

Such a topology is called a *cloud* and includes protocols such as Asynchronous Transfer Mode (ATM), Frame Relay, Switched Multimegabit Data Services (SMDS), and—less commonly in the US—X.25. Packet-switched connectivity is ideal for organizations whose WAN traffic is “bursty” or variable in nature and does not require strictly dedicated bandwidth or always-on WAN links.

WAN Virtual Circuits

So-called *virtual circuits* may be defined as any logical circuit created between two network devices across a shared network medium. There are two types: switched and permanent virtual circuits.

Switched virtual circuits are dynamically established on-demand and terminated once interaction between the two linked parties ceases. Three phases define a switched virtual circuit’s operation: circuit establishment (connect parties), data transfer (exchange information), and circuit termination (end conversation). This setup is analogous to making a telephone call, which follows this sequence: dial the other party and establish the call (connect parties), conduct a conversation (exchange information), then disconnect the call (end conversation).

Once initially established, permanent virtual circuits (PVCs) remain locked into data transfer mode thereafter for an indefinite period of time. (Though such circuits can and do go down occasionally for various reasons, they are considered “always up” by virtue of their operation.) PVCs are utilized in situations in which data exchanges are constant and ongoing between two points. These circuits involve increased costs for usable bandwidth owing to their constant availability, but that bandwidth also comes with availability and reliability guarantees.

WAN Dial-Up Services

Dial-on-demand routing and dial backup are two popular and cost-effective dial-up services for WAN connectivity. Dial-on-demand dynamically initiates calls when data is ready to be sent and specifies an idle timer that disconnects after a specified period of inactivity.

Dial backup uses switched circuits to provide backup services for a primary circuit such as point-to-point or packet switching. Routers are configured for failure detection and automatic failover until a disrupted primary connection is restored.

WAN Devices

A typical WAN comprises numerous networking devices, most of which are not unique to the WAN environment itself. Modems, switches, and servers are non-specific, general-purpose elements in every business computing landscape. These devices bridge network connectivity among LAN and WAN segments, where each type provides different advantages and benefits, along with individually applicable disadvantages and drawbacks. Let's examine each representative category in turn.

WAN Switches

Typical LAN-based Ethernet switches are multiport networking devices used in localized environments. Similarly, WAN switches perform identical functions for distributed networking contexts. They operate at the data-link layer (OSI Layer 2) and switch traffic from Frame Relay and SMDS.

Access Servers

Central dial-in/dial-out gateways for dial-up connections are called *access servers*. These devices provide LAN and WAN networking equipment access to asynchronous devices. Network access servers function as control points for roaming and remote users so that they may access internal resources (or to an ISP) from external locations.

Analog Modems

An *analog modem* translates between analog and digital signaling. This enables data-bearing communications to transmit via voice-based telephony. Digital signals are converted into an analog format suitable for transmission through analog carriers and then restored to digital format on the receiving end.

Channel Service Unit/Data Service Unit

Digital phone line subscribers are connected to telephone service provider network equipment through *channel service units* (CSUs). End-user equipment (for example, routers, computers) interfaces this modem-like device to access network resources through the provider's local digital telephone loop such as a T1, E1, or DS-3.

The device that connects CSU to data terminal equipment (DTE) is called the data service unit (DSU). It adapts the physical interface on the DTE to the provider transmission facility (for example, E1, T1, DS-3) that converts between subscriber and provider protocol formats.

ISDN Terminal Adapter

An *ISDN terminal adapter* is like a modem in that it joins Basic Rate Interface (BRI) connections to different physical interfaces on a router. Unlike a modem, it does not convert between analog and digital signaling.

Understanding the WAN Optimization Landscape

Because of the many types of WAN links in use, there are likewise many challenges to and best practices for making the most of WAN optimization, where no two vendor solutions and strategies are exactly alike (though most share numerous elements in common). Some optimization techniques are incremental and provide only marginal performance improvements for application-specific traffic flows. Other optimization solutions are instrumental to sustaining long-term performance-enhancing goals and can fundamentally change the way certain network protocols operate—most of which occurs transparently between the application and its end user. In addition, some vendors provide different classes of accelerated performance in their optimization products, including such major enterprise applications as Microsoft Office SharePoint, SAP, and Oracle Financials.

One key aspect in accelerating fundamental WAN performance comes from changing undesirable behaviors, primarily by eliminating excessive, repeated, and wasteful transmissions. The most efficient strategy in achieving this goal comes from avoiding unnecessary data transmissions altogether. Data caching, data compression, and data reduction are three techniques that haven’t proven able to provide measurable benefits in this regard. Data caching, data compression, and data reduction strategies have been discussed in preceding chapters, so their redefinition here is unnecessary. Extensive acceleration strategies go well beyond these concepts, but WAN optimization absolutely involves multiple forms of data caching, compression, and reduction.

📖 See Chapter 2 for more information about data substitution, caching, and compression.

Actual data reduction implementations and methods vary widely among vendors and product platforms. For the purposes of this chapter, it suffices simply to distinguish among distinctive differences between data caching and data reduction approaches (data compression is completely different and mutually independent).

Data Reduction and Data Caching Compared and Contrasted

Data reduction has the following advantages over data caching:

- Application breadth—Reduction strategies detect patterns across diverse types of traffic, whereas caching takes an application-specific object-level orientation. Data reduction is an endpoint-oriented technique that reduces traffic volume by sending placeholders in the absence of duplicate data (it gets restored and reissued on the other side). Byte-level granularity detects higher resolution by indexing blocks of patterns in network traffic even when an application protocol (such as backup or replication) performs similar functions at another level (especially at the file, block, or object level). When repeated sequences are replaced in data flows, corresponding index or dictionary references—not repeated data elements—are sent across the WAN link. This approach offers sometimes extraordinary reductions in data volume.
- Application transparency—No client-server modification is necessary to employ data reduction methods, but some caching environments require individual endpoint configurations for all participating clients.
- Coherency—Preservation of client-server communications eliminates the chances of delivering stale or inconsistent information when employing data reduction strategies. Maintaining cache currency and coherence can involve significant amounts of processing activity and WAN communications.
- Effectiveness—Data reduction operates at the byte level instead of the object level for data caching techniques. This offers a more granular, higher-resolution hit rate when detecting duplicate information, including information contained within apparently different objects.

Both data caching and data reduction employ a common data management strategy: that is, both depend upon a central point of access that also acts as the controlling authority for endpoint-to-endpoint transactions. Despite vast differences among vendor platforms and products, there are several noteworthy aspects common to all such implementations:

- Long-term timeline storage—Highly effective data reduction strategies leverage native drive storage housed in network appliances to maintain historical snapshots of application and protocol behavior. Access to several months of traffic patterns and data efficiently eliminates duplicate data delivery over the long term by recognizing extensive recurring patterns that may otherwise be missed in shorter-term trend analyses.
- Effective capacity—Vendors differ greatly in the methods they employ to parse and store data as part of a data reduction strategy. Some are more efficient than others, and make more effective use of available storage. Bidirectional solutions operate using two-way communications and can optimize flows in both directions, whereas unidirectional strategies must manage data separately for every individual traffic flow.

- Application breadth—Data reduction solutions operate at the network layer of the TCP/IP network stack to support any transport protocol including UDP. Solutions that specifically target TCP flows are designed to footprint and store bulk TCP application data (such as file transfers and email messages). Support for UDP streams expands the breadth of supported applications (including VoIP as used for IP Telephony and related services, and the Real Time Streaming Protocol—RTSP, as used for streaming media playback over the Internet, primarily for entertainment videos).
- Data protection—Data reduction solutions take protective measures to safeguard end-user information that usually involves application of encryption mechanisms. Compression and reduction strategies work well on repetitive data elements, but effective encryption randomizes such data and renders those strategies ineffective. SSL acceleration originating and terminating on the WAN optimizer expedites overall traffic by permitting optimization mechanisms to operate even within encrypted (therefore unintelligible) transmission streams (essentially, this involves sharing keys or certificates, decrypting data streams in the device to seek out repetition, applying data reduction and caching techniques, then re-encrypting the reduced output for transmission across the WAN. The benefits of WAN optimization usually outweigh the associated overhead involved, making this approach entirely cost effective).
- Granular matching—Each solution also differs in how it seeks matching data patterns both in the granularity of the search employed and the resulting long-term database fingerprints. Some solutions work well for duplicate data strings or streams sent in rapid succession but may be ineffective when working with derived data or duplicates sent after older data ages out of the cache.

Finally, data compression seeks to reduce traffic traversing the WAN topology. Simple algorithms identify repetitive byte sequences within a single packet, whereas more sophisticated implementations go beyond the packet level to match packet sequences and entire protocol streams. Header compression provides further bandwidth gains through specialized algorithms designed for protocol-specific properties. Payload compression algorithms identify relatively short byte-pattern sequences in data-bearing protocols that recur over a measured duration, which are replaced with shorter references. Compression across various flows of traffic is called *crossflow compression* and works even on UDP-based traffic.

In each of these strategies, a centralized analysis and control point is required to monitor and modify entire network transactions through individual conversations. The proxy appliance or proxy server dutifully services this role and proves itself a greatly effective tool in enhancing performance and security capabilities for given client-server needs.

WAN Optimization Delivers the Goods

Through a consistent notion of where and how data travels, what is repeated (and thus, may be copied), and which applications (and identities, services, and so forth) are utilizing WAN links, WAN optimization devices create a model for networked communications that lets them maximize the traffic that WAN links can handle, and make the most of the WAN resources and bandwidth at their disposal. Enterprises and organizations also quickly realize that these devices provide a "golden opportunity" to impose and enforce policy and police network usage at the very chokepoints unwanted traffic might otherwise overwhelm. It's rare that technology solutions pay off more handsomely than they're expected to do so, but WAN optimization enable savvy buyers to put their money right where the pain and pressure usually reside, and realize a veritable bonanza as a result.

In the fourth and final chapter of this e-book, we will dig into the challenges most likely to be encountered when implementing WAN optimization, and the best practices that organization and enterprises should heed when deploying WAN optimization solutions. By the time you finish that material, you should have some good ideas about how to respond to those challenges, and what best practices you'll want to implement as well.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit http://nexus.realtimepublishers.com.