

# DELIVERY, PLEASE!

In this executive interview sponsored by Citrix, Burton Group's ERIC SIEGEL reveals the best practices behind application delivery.

## **Application delivery infrastructure is growing rapidly in strategic importance for many CIOs. What are some of the broader market trends driving this change?**

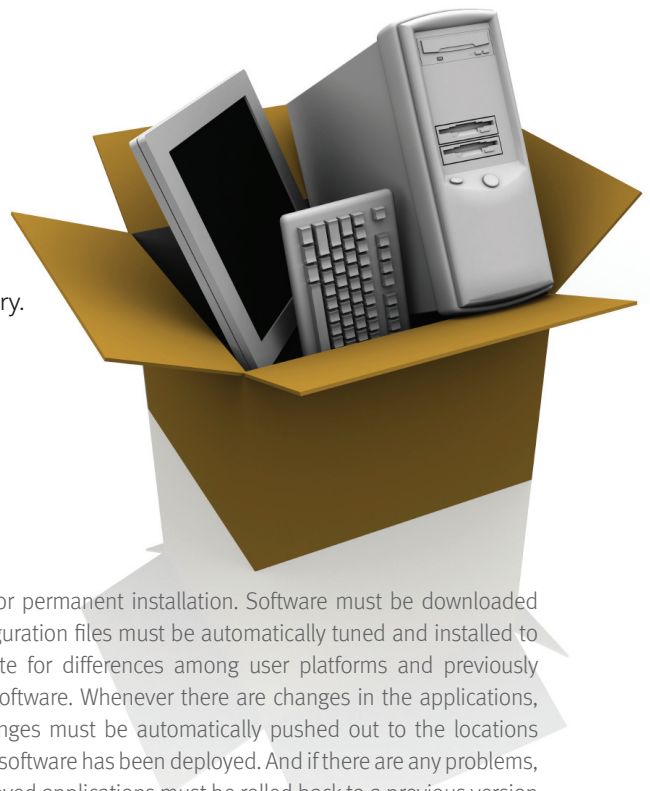
For years, enterprises have moved toward decentralizing applications. Those applications in which both the application and most of its data resided close to the user could provide excellent performance because there was almost zero latency and massive bandwidth interconnectivity among the application, the data and the user. Tightly-coupled designs in which a single transaction might require dozens or hundreds of interactions between the user and the application and between the application and the data weren't a problem. Performance was good and it remained good regardless of the latency or bandwidth of the communications link between the user's computer and the central enterprise facility. (And although there could be problems when updating the data on the user's computer, those updates could often be scheduled outside of work hours.)

Recently, however, there has been a strong move to centralize applications in response to security and legal requirements, and because enterprises discovered that maintenance of a large, dispersed community of users was expensive and difficult. Ensuring that all those remote applications were backed up and audited was a challenge. Not only were backups lost or not performed, security audits were difficult and sometimes required auditors to travel to remote sites. New legal requirements increased the penalties for mistakes, the cost of remote operation and competent staffing of these distributed systems rose. Centralized servers began to look more manageable and less expensive to many IT managers.

A centralized architecture usually has much greater latency among the application, its databases and the user than a decentralized architecture. Worse, when a user travels, thereby varying the latency and bandwidth to the central servers, the performance of the application can change massively. The number of back-and-forth interactions among user, application and data serves as a multiplier for any latency or bandwidth problems. If it takes 50 back-and-forth interactions to open a file, and the latency between user and central server increases by a factor of 10 when the user travels to a remote location, then the performance of the application may suddenly be 500 times worse. The variability in performance can sometimes be just as irritating as the absolute delay. Therefore, it is not surprising that users are complaining or that the architectures used for delivering an application to users are becoming more visible to IT management.

## **Most IT organizations talk about "deploying" applications. How is the notion of "delivering" applications different and why should it matter to CIOs?**

The term "deploying" can imply a complex process of provisioning a remote application by sending its databases, configuration files and interconnecting communication paths to the remote user



platform for permanent installation. Software must be downloaded and configuration files must be automatically tuned and installed to compensate for differences among user platforms and previously installed software. Whenever there are changes in the applications, those changes must be automatically pushed out to the locations where the software has been deployed. And if there are any problems, then deployed applications must be rolled back to a previous version or uninstalled without creating more problems.

In contrast, if an application is thought of as a service, not as a set of software, then the application's function can be "delivered" either by "deploying" it, when that's the best method, or by providing some other method of accessing the application's services, such as by use of a Web browser, a thin client or Web services.

Thinking in terms of "delivering" instead of "deploying" has the effect of helping the CIO think about the problem of user access more broadly, which can help find new solutions to difficulties.

## **As the employee workforce grows increasingly mobile, what are some of the best practices to ensure that application users have fast and secure access to their applications at all times?**

Ideally, applications would be designed to be loosely coupled without a great deal of dependency on low latency or high bandwidth interconnections to remote servers. Unfortunately, that's not practical in most enterprises that need to handle large numbers of legacy applications.

Thanks to the widespread availability of public applications like YouTube, users really prefer and are actually beginning to expect that they will be able to use the same procedures to connect to an application regardless of where they are – and that the connection will always be secure and provide high performance. Today, users expect to be able to use the same procedures whether they're in the central office, a branch office, at a customer's site, in a hotel room or on a cellular connection. Therefore, IT managers should work toward a unified system that includes both secure access and protocol acceleration, or other types of application delivery such as thin clients.

It is also important to consider how different optimizations interact. For example, encryption can interfere with protocol acceleration and data compression, and some types of compression may interfere with other types of compression. If an organization needs to encrypt data traffic, it must ensure that either the optimization and compression functions have access to the data stream before and after that

encryption, or that those functions must be given the appropriate keys to examine and manipulate traffic. Interoperation between the optimization and compression functions and any firewalls is also important because optimization and compression may interfere with firewall operation. Suspicious firewalls may then shut down optimized, compressed traffic.

If protocol acceleration and compression are built into, for example, a user's laptop, then those functions must be workable both when the user is connecting directly to the central servers and when the user is connecting through a branch office's backbone that may contain its own set of traffic acceleration and compression appliances. (For example, the common "zip" procedures used to compress browser traffic can conceal data duplications that advanced compression devices would otherwise be able to remove from the data flow.)

In all cases, end-to-end and intermediate measurement at system demarcation points is crucial for both the helpdesk and for the system operations center. That measurement must work despite any encryption, optimization or compression functions, and if necessary, it must help the operations staff quickly isolate any difficulties to the responsible organization for further analysis.

**Application virtualization solutions enable companies to centralize all of their Windows applications in the datacenter and deliver to the client by virtualizing or streaming them to the desktop. What are some of the advantages of this approach to delivering client-server and desktop applications?**

Centralized management of the Windows applications may be simpler than distributed management, although organizations with tight control of user platforms and applications can provide similar abilities and may provide better performance in some situations.

**Web applications are centralized in the datacenter by definition, yet IT organizations face similar challenges around delivering them to end-users with great performance, security and cost savings. What kind of infrastructure solutions should companies have in place to ensure the fast, secure delivery of Web apps?**

A number of vendors provide acceleration and compression functions in the application front-end (AFE) appliances used to provide load balancing and other front-end processing at the central datacenter. These functions can work without any changes to the user's computer system, or they can work with client-side optimization software that has been installed directly in the user's computer. (That can help in environments where it's impractical to add a wide area network [WAN] optimization appliance to the user's side of the communications path.)

Client-side software either works with a browser, optimizing data flows through the browser or to applications that use the browser's

hypertext transfer protocol (HTTP) and file transfer protocol (FTP) stacks (Microsoft Internet Explorer's Windows Internet [WinInet] interface), or it works with software that's installed into the system and outside of the browser where it can optimize all types of Internet protocol (IP) traffic. Browser software can be inserted automatically by the server system as an Active-X control, JavaScript or Java, for example. Other client-side software must usually be installed manually or through an enterprise's standard updating mechanism, but can then intercept and optimize data flows that use the Windows Socket (WinSock) library or other methods that aren't based on browsers.

There are a number of optimizations that can be performed by software in the user machine in addition to the standard optimizations of compression, improving error and flow control, and also improving access to remote files. The connections between the client and server can be optimized, cache effectiveness can be improved and some "read ahead" can be performed to pre-load files into the client before they're requested by the client application.

Some applications and Web pages may repeatedly open and close transmission control protocol (TCP) connections incurring overhead on the WAN and in the endpoints. Optimization software in the business can recognize that behavior and hold connections open or multiplex parallel short-lived connections between the same pair of processes into a single, persistent flow. That single, persistent flow can then be spread among multiple persistent connections across the WAN, resulting in higher effective throughput. Trusted client optimization software can also create credible quality of service (QoS) prioritization tags for outgoing packets.

Cache behavior can be improved by optimizers. The pair of client-side and server-side optimizers may be able to manipulate the browser's cache and its use of caching including its use of HTTP cache control tags. That can improve response time for fetching files that are already inside the client; there isn't any need to send even a single round-trip query/response across the WAN if the caching mechanism is optimized. (For example, the freshness of all of a Web page's objects can be validated at the same time, possibly when the initial base HTTP file is fetched, instead of with separate queries.)

In addition, the server-side optimizer can act as a sophisticated server-side ("reverse") cache, offloading the servers and concurrently improving response time – even for some files that would not normally be cached in a server-side cache, because they might appear to be dynamic to an unsophisticated cache. The server-side optimizer can also scan outgoing HTTP for embedded names of other files that will probably be loaded later, then push those files into the client cache before they're requested. Many of these optimizations can be performed even if there isn't any special software in the user's client.



ERIC SIEGEL, a Senior Analyst at the Burton Group, covers Web and network performance optimization, measurement, management, SLAs and QoS. He has been a member of the Internet community since 1978 and is the author of "Designing Quality of Service Solutions for the Enterprise" (John Wiley & Sons) and "Practical Service Level Management: Delivering High-Quality Web-Based Services" (John McConnell with Eric Siegel; Cisco Press). Before joining Burton Group, Eric was the Principal Internet Consultant at Keynote Systems, and he was a Senior Network Analyst at NetReference Inc., where he specialized in network architectural design for Fortune 100 companies. Previously, he was the technical leader and coordinator for all of the data communications specialists at Tandem Computers.