Restoring User-Application Linkage

Fixing Unintended Consequences of Business Initiatives

By Peter Sevcik and Rebecca Wetzel
August 2005

Change has unintended consequences, and an unintended consequence of five widely embraced current business initiatives is to undermine the performance of networked applications. These pandemic changes – the decoupling of software into modular elements through object-oriented design and service-oriented architecture (SOA), the atomization of organizations to make staff increasingly distributed and mobile, globalization, increasing electronic outreach, and data center consolidation – make business sense, but can so debilitate application performance that the new business initiatives themselves are jeopardized.

To ensure networked applications perform up to par, IT managers are turning to a new class of solutions, whose goal is to compensate for the unintended consequences of these changes as well as others. The solutions form an **application delivery system**, which measures, controls, and improves the performance of networked applications to ensure that they meet business needs.

Application Delivery Systems routinely deliver 3X throughput increases and 10X response time improvements.

NetForecast Report 5078

Why Is Performance Eroding?

Information applications are the lifeblood of today's business, and depend on strong links between users and applications. The degree of linkage between applications and users, and among application elements influences performance. Peak performance requires that the linkage among what networked applications do, where they operate, and who has access, is well managed. Traditional IT management practices were based on strong user-application linkage. Managers had an intimate knowledge of their applications and the users they served. Unfortunately, linkage becomes increasingly challenging as the following well-intentioned and valuable business initiatives take hold – and weak linkage adversely affects performance.

Software Decoupling: To make software easier, faster, and cheaper to deploy, developers are flocking to new standards and development methods that decouple software elements. For example, object-oriented design decouples software into modular elements or "objects" that can be re-used by different programs without being re-tested.

Similarly, service-oriented architecture provides a loosely coupled way for different applications running on a variety of platforms to interoperate, and enables existing IT systems to be re-used to create new services. Web services, created using XML and its many cousins, are examples of SOA in widespread use. These decoupled services provide business functions used by clients in different applications or business processes.

Object-oriented design and SOA decouple not only application processing, but also storage, data, and, ultimately, the user. The upside is cost savings, flexibility, and speed of deployment. But the inevitable downside is performance degradation.

Organizational Atomization: The drive for business agility combined with the ubiquity of broadband, are nudging organizations away from a centralized, hierarchical structure to a flatter, more distributed organizational structure with many remote offices, increasing numbers of mobile tele-workers, and stronger ties to strategic partners. While increasing flexibility, productivity, and customer responsiveness, atomization makes it hard to manage application use, hurting application performance.

Globalization: For most large businesses, the marketplace is becoming the world. The drive to procure materials and/or sell products worldwide witnessed quantum leaps with the introduction of the steamship (1819), trans-oceanic flight (1927), international direct telephone service (1965), and global electronic mail (1989). The relentless

interconnection and interdependence of global markets continues, further fueled by global outsourcing and overseas competition.

Although essential for most large businesses, globalization stretches business applications designed to run over limited distances and uncongested links, to span the globe and serve regions with severely constrained links. These performance-degrading situations transcend any IT manager's reach.

Electronic Outreach: Cost and convenience are propelling audio and video communications to the IP network. Video conferencing is becoming a necessity for intra-corporate communication, and Web-based, rich multimedia is catching on for communicating with and training customers and business partners. Electronic outreach is a powerful business tool, but it is difficult to assure a satisfactory experience for users separated by vast distances, and/or located behind congested or low-speed links.

Data Center Consolidation: Conformance with regulations such as the Sarbanes-Oxley Act in the United States or software licensing compliance, combined with cost savings enabled by advances in server virtualization, are driving businesses to centralize data centers. Although regulatory conformance and cost saving are compelling benefits, shifting servers away from users challenges IT to deliver satisfactory application performance.

Why Performance Matters

Performance is important because customers, employees, and partners must be satisfied in order for the business changes described above to succeed. Yet, by their nature, these initiatives can dramatically degrade IT performance, slowing or even halting adoption. A satisfactory user experience is a prerequisite to success, and end-user response time is the canary in the application performance coal mine. If end-user response time is satisfactory, then proper linkage is in place. If it is not, then the business initiative may be at risk.

The response time for transactional applications depends on four factors described by the following formula.

$$ResponseTime \approx Turns(RTT) + \frac{Payload}{Bandwidth}$$

Where:

ResponseTime is defined as the elapsed time (seconds) between a user action (e.g. mouse click, enter, return) and the system response (client, network, server), so the user can proceed with the process. The aggregation of these individual task completion waiting periods defines the "responsiveness" of the application to the user.

Turns are the application client-server software interactions (turn count) needed to generate a user-level system response or task (see above). The user is not aware of turns.

RTT is the round-trip-time (seconds) between the user and the application server.

Payload is information content (bytes) that must be delivered to/from the user's device.

Bandwidth is the minimal bandwidth (bits per second) across all the network links between the user and the application server. The slowest link is typically the user's access line to the network. Bandwidth may be reduced by the effects of conflicting traffic (congestion) and protocol efficiency (e.g., TCP window).

Each of the five business changes described above increases application response time in the above formula. Here's how:

- *Software Decoupling* adds turn count through application complexity such as the need to discover application elements. The unending growth of application complexity increases Turns-per-Task at a relentless pace.

- *Organizational Atomization* decreases the user's access bandwidth (e.g. moving from corporate headquarters to a remote or home office often moves employees from a high-speed connection to dial-up or wireless connections), and the further the user is from corporate locations, the longer the RTT.
- *Globalization* decreases the user's access bandwidth (and may add packet loss), especially for users in locations with limited infrastructure. But regardless of infrastructure, connecting to distant locations increases RTT to all users in those locations. Also, the speed of light becomes a significant factor between continents. It is not unusual for global RTT to consistently reach 0.5 second.
- *Electronic Outreach* adds payload as applications supply more information such as help screens and multimedia.
- *Data Center Consolidation* increases RTT to users distant from a central data center.

The cumulative effect of these response-time challenges is that applications often hit the wall after they are deployed, with devastating consequences to the business initiative. As a result, a new corporate strategy may be abandoned outright, or require much more time, effort, and money than anticipated to succeed.

This is, of course, not a new phenomenon. Each previous wave of computing and application design innovation has faced similar challenges, which were subsequently solved by IT infrastructure. For example:

- Host-based applications required host processing and memory engineering to succeed.
- Client-server applications needed significant bandwidth across corporate networks to operate correctly.
- Consumer-focused Web-based applications drove broadband access (cable modem, DSL) to the home.

Unfortunately, faster computers and/or more bandwidth will not solve the application performance challenges caused by these five business initiatives. What is needed is a new class of solutions.

How to Restore Linkage and Improve Performance

An emerging class of Application Delivery System (ADS) solutions strengthens linkage by acting on the flow of data between server and client. These solutions measure, control, and improve the performance of networked applications. Dozens of ADS techniques are widely deployed to solve linkage-based performance problems through intelligent policy-based intermediation between users and multiple application elements. These solutions routinely deliver 3-fold throughput increases and 10-fold response time improvements.

When a new application is initially deployed at or near corporate headquarters, for example, performance is generally excellent. The consequences of application complexity, network distance, and constrained bandwidth only surface when the application is deployed to remote users. These adverse consequences can so damage performance that users abandon vital applications in frustration, causing business initiatives to fail. Application delivery systems mitigate the effects of complexity, distance, and bandwidth constraints, often delivering LAN-like performance globally.

To illustrate the before-and-after effects of ADS deployment, NetForecast employs an open standard user satisfaction scale called Apdex (Application Performance Index). The following figure shows a prototypical example of how an application delivers excellent performance when deployed at headquarters, poor performance when deployed globally without the benefit of ADS technology, and excellent performance once again when ADS

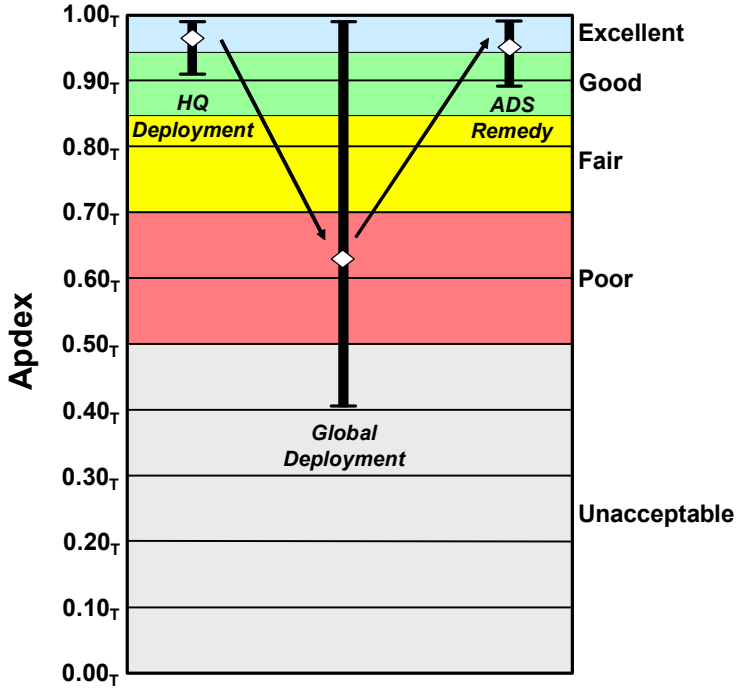
Example ADS Vendors:
Akamai
Blue Coat
Cisco/Fineground
Citrix/Netscaler
Data Power
Expand
F5
Juniper/Peribit
Juniper/Redline
Netli
Packeteer
Riverbed Technology
Sarvega
Solace Systems
Tacit Networks

Apdex:
Apdex is an open performance reporting standard defined by the Apdex Alliance. See: www.apdex.org

NetForecast helps enterprises and vendors understand and improve the performance of networked applications.

Additional information is available at: www.netforecast.com

technology is introduced to mitigate the unintended consequences of globalization. The I-bars represent the range of Apdex values across all locations while the diamonds show overall performance for the enterprise.



Note that while the Apdex score shifts from excellent to poor after global deployment, the Apdex performance range (best-to-worst served users, indicated by the I-bars) remains excellent at headquarters, thus masking the impact of the business changes to corporate management.

A great benefit of ADS technologies is that they improve performance by restoring user-application linkage in ways that are transparent to the user as well as the application. They form a new, much needed, infrastructure layer that operates as an intermediary between applications and their users to measure, control, and enhance user sessions. This new layer in the IT architecture can be implemented in the normal course of IT upgrades and because ADS solutions are independent from the application and the user, they tend to be easy to implement, and relatively inexpensive.

If you are embarking on one or more of the business initiatives described here, you should develop an ADS strategy and implementation plan. You will find it an essential investment in the success of your IT strategy.

Peter Sevcik is President of NetForecast and is a leading authority on Internet traffic, performance, and technology. Peter has contributed to the design of more than 100 networks, including the Internet, and holds the patent on application response-time prediction. He can be reached at peter@netforecast.com.

Rebecca Wetzel is an Associate of NetForecast and a 20-year veteran of the data networking industry, with unparalleled inside knowledge of the Internet service and product markets. She works with network product vendors and service providers to develop and implement strategies for new IP-based offerings. She can be reached at rebecca@netforecast.com.