Restoring User-Application Linkage

Net Forecasts – Peter J. Sevcik
BCR Volume 35, Number 9
September 2005

I know a nice old gentleman who owns a great piece of land in a suburb of Boston. As he was scaling back his carpentry business, he was planting and tending to a wide variety of pine trees – future Christmas trees to be exact. He started the business one fine fall, telling me all about his plans and hopes for a great business. He would sell his trees to a few good customers. They would spend a lot of time in his fields looking for that perfect tree that they would cut down themselves. He would talk to the family and learn about their holiday plans, family visits and that important gift this year. In other words, he would have a great relationship with his customers that would develop into a lasting customer-business linkage.

Things started off well. I was one of his first customers. I knew him well before the doors opened so we already had good linkage. He built a log cabin in the middle of the field. Had a fire in the stove all the time. Hot cider for free. Nice place to sit, talk, reminisce, see old friends, think about good values, and come home with that perfect tree. He was connected to his customers.

By the second year, the dream started to fray. Customers got to pick a tree well in advance of Christmas, tag it, and cut it just as they needed it to arrive in their home. However, others would come along, remove the tag, cut down the tree and take it. The original customer was devastated when they arrived days before Christmas to see their tree gone and no good tree left to buy.

Things kept getting harder for the proprietor. Business grew. People wanted more trees. They were not so interested in the variety he had. Everyone wanted a 6-foot fir. So he started to import them cut from Canada. In the next few years he transitioned from a service to a product business.

Then came the help he had to hire. His cabin turned into an office for the staff. He had to add security to keep people from stealing trees. The business grew in revenue but he made less money. He let the "mercenaries," as he called them, run it all while he stayed in the old farm house.

At the end of that season, he closed the business because he no longer talked to any customers. He lost all customer linkage. No point keeping it going. He still has dreams of a tree farm where he and his customers know each other. He owns a good parcel of land in Maine where he is again growing trees.

How User-Application Linkage Is Lost

There are two lessons to learn from the Christmas tree farmer: You must maintain good customer linkage; and it is easy to lose that linkage.

If you are an IT manager, you must maintain user-application linkage. In the old days this was easy. You knew who your users were, where they worked, and what they did. You even talked to them in the cafeteria at lunch where you learned about system issues and problems first hand. Those days are gone for good.

The user-application linkage was lost in two phases: control, followed by context. In IT, this can be restated as the unintended consequences of adopting open standards, followed by adopting competitive business initiatives. All the changes described in this column follow sound technical and business practice. The unforeseen loss of control and context is what must be tracked and then reversed by other means.

Phase 1 – Loss Of Control

Enterprises have embraced open standards and the Internet. Gone are the days of proprietary hardware IT architectures. Every major software vendor is converting its software to Web open standards. These are good initiatives. Enterprises gain vendor independence, have product choices, and can buy from an open market that supplies innovation at a fast pace – all for less money.

But a price is exacted, in loss of control and visibility into the user-application connection. Table 1 uses the NetForecast Performance Functions to compare the old closed way to the new open way by which users connect to applications. (These functions were originally described in BCR,

November 2003, pp. 8-10).

The many changes can be restated as a shift from enterprise initiation of the connection to user initiation. Enterprise IT no longer finds the user; the users find the enterprise on the Internet. This is just like shifting a store from having a sales clerk find you the shoes you need, to the store going self service with open stock. However, you will notice that self-service shoe stores are very particular in using those annoying nylon ties to make sure each pair of shoes stays together. Why? Because they lost control over the shoes.

Table 1 shows a clear loss of control. When you move to the efficient open Internet model of connecting users, you no longer know who, where, or how a user is accessing your system. And you certainly don't know what else they may be doing with the device that they are using to communicate with your computers.

Table 1 – Comparing Closed and Open IT Strategies

	Old Way	Current Way
Provisioning The overall service uptime, speed of recovery, and the ability to add features seamlessly. AKA System Availability.	● IT installed client software and locked down the PC	● Everyone has a browser and can create their own apps easily.
Efficiency The ability to maintain high utilization of assets, driving down the cost of operations, and resulting in good ROI. This starts with good visibility.	● IT authorized CIR, storage, etc. ● Network staff sees all traffic by unique TCP ports	● Everything looks like port 80 Web traffic to the network ● Even worse if it goes SSL
Protection The ability to detect and stop denial of service attacks, virus infections, and worm propagation. AKA System Security.	● Can't find me or see me - security by obscurity (proprietary systems) ● Users can't do anything else with the application	● Browser has many vulnerabilities ● Lots of fun/bad things to do with a browser
Accessibility The ability for the user population to access the service regardless of their means of connection or location. AKA uniform access.	● Connection to remote office implies access for all those users ● Can ping each desktop	● If they don't call help desk or customer support, then they must be getting through
Quality The ability to provide satisfactory user response time (system responds to a user query in sufficient time so the user is satisfied with the application's responsiveness). AKA speed.	● Built into the "thick" client ● Some have performance instrumentation	● Inefficient chatty "thin" client ● No performance instrumentation
Safety The ability to ensure that there will be no additional traffic from the service (e.g. no virus, spam, spyware) and that the user is kept safe when interacting with the service (e.g. assurance of user privacy, identity protection, credit card information safeguards). AKA trust.	● The user is "hardwired" to the service. ● The application can't do anything else to the user.	● Loss of trust via spam, spyware, viruses, phishing, identity theft, etc

Phase 2 – Loss of Context

Context is the circumstances in which an event occurs. Your business applications are being delivered to users who operate in differing circumstances. Initially this is not a problem. Although your users are now connected in a new way, you still know who they are and what they do. The user population or its basic behavior does not change overnight just because they have new access.

For example, many users had email and Web access on their office desktops in the mid 1990s, but did not know how to use these programs. The widely publicized Web changed all that as the users helped themselves to the new medium.

So it takes more than cutting ties to the user to lose context. The business or the user must actively do something that causes them to drift apart. And the way you know that the drift is severe is to watch application response time.

Enter five current business initiatives:

- 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.
- Datacenter consolidation.

These pandemic changes make business sense, but can so debilitate application performance that the new business initiatives themselves are jeopardized. Unfortunately, as the five business initiatives take hold, linkage becomes increasingly frayed, in the following ways:

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, is nudging organizations away from a centralized, hierarchical structure to a flatter, more distributed organization with many remote offices, more mobile teleworkers, and stronger ties to strategic partners. Atomization increases flexibility, productivity and customer responsiveness, but 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 made quantum leaps at key points in the last two centuries: 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 continues, 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 into the IP network. Videoconferencing is becoming a necessity for intra-corporate communication, and Web-based multimedia is catching on for communication and training with customers and 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.

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

Performance Shows The Loss Of Linkage

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. Bandwidth may be reduced by the effects of conflicting traffic (congestion) and protocol efficiency (e.g., TCP window).

Each of the five business initiatives 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 HQ to a remote or home office often moves employees from a high-speed connection to dial-up or wireless connections), and the farther 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 half a second.

Electronic Outreach adds payload as applications supply more information, such as help screens and multimedia.

Datacenter Consolidation increases RTT to users distant from a central datacenter.

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 adds up to five ways in which you are pushing away from your users, thus losing context. In the end you do not know where, what, how, or why they are using your applications. What are the chances that your enterprise is not losing user-application linkage?

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 in order 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 business initiatives. A new class of solutions is needed.

Restoring Linkage, Improving 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 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. ADSs 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, we employ an open standard user satisfaction scale called Apdex (Application Performance Index, see, March 2005, pp. 8-10). Figure 1 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 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.

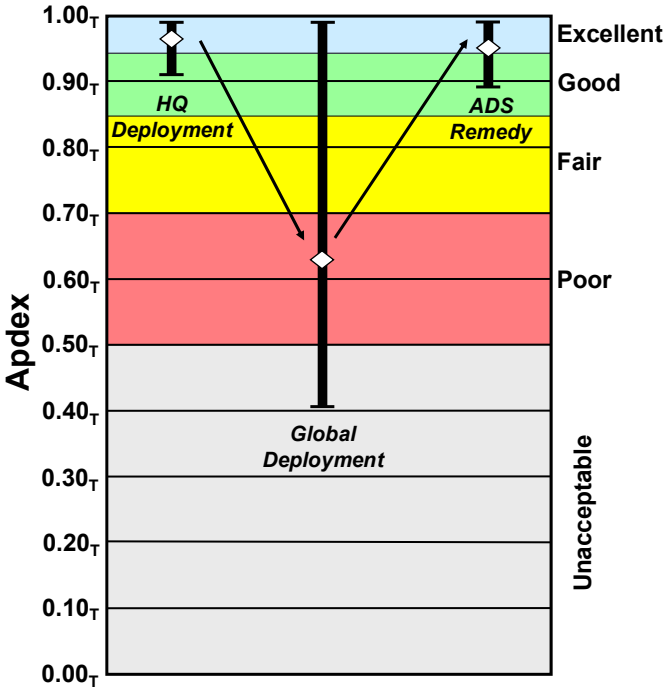

Figure 1 – The Beneficial Effect of ADS

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 corporate management's view of the impact of the business changes.

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, measuring, controlling and enhancing 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 of the application and user, they tend to be easy to implement and relatively inexpensive.

Long-Term Solution Component: Session Protocol

Clearly, ADS technologies provide a benefit that addresses the more immediate second phase of poor user-application linkage: loss of context. But long term, there will be a need to address the first phase of how we got into this mess: loss of control. To deal with this issue, we need to go back to some fundamentals.

Long ago, as the packet revolution was starting and people were trying to find an interoperable way to communicate among business systems, the ISO defined a seven-layer protocol architecture that would encompass all functions needed. The architecture is still quoted by IT professionals, but they rarely look at where we are in implementing the original vision.

The seven-layer vision was good; but the new standards defined by ISO were too complex and tried to re-state many of the old proprietary host-based systems in an open way. In the meantime, a new crop of open standards based on the Internet emerged as the winning protocols everyone uses (Table 2).

Table 2 – The ISO Seven-Layer Model Today

Layer	ISO Name	Open Standard	Leading Standards Body
7	Application	XML, SOAP, etc.	W3C
6	Presentation	HTTP, HTML, etc.	
5	Session		
4	Transport	TCP	IETF
3	Network	IP	
2	Data Link	Ethernet	IEEE
1	Physical	Many	

The real news and action in the 1970s was not in the upper protocols, but the fact that layers 1 through 4 appeared for the first time. Believe it or not, these were all just one layer in the old host systems – link protocols did it all. The IETF jumped into this new area with many standards required to grow the Internet. Of course, the IEEE was equally active with its Ethernet variations.

The next wave of open innovation appeared in the 1990s, with the Web and then the new W3C standards efforts. So the ad hoc approach to the ISO model came at it with separate initiatives from the bottom and the top. These are now the de facto standards for all IT projects. Everyone is calling it a great success story.

Not so fast. Notice that this dual-pronged approach left the session protocol conspicuously un-implemented. Both groups thought this was the one protocol that came from the old host-based thinking that could be ignored. But now that we have lost the user-application linkage, isn't this exactly the protocol we need, to regain control?

We need a session protocol to solve the mapping of many-to-many ADS techniques. This is what we need to let the ADS solutions scale past solving issues for a single enterprise. This is where the user is bound to the application on an end-to-end basis with uniform definitions of services, priorities, SLAs, etc. We can learn that this is a legitimate user that has passed some security screening in a way that can be exchanged among business partners. We can now learn about the user context

(this user is on a poor wireless connection, but is one of our best customers) and match it to the application delivery solution required for that application (but he is only browsing our catalog and not filling a shopping cart).

Here we can also define and ensure user experience parameters with an open standard like Apdex. The session can be monitored and reported on through an extensive session profile.

It is time to revisit the session protocol as a way to bridge the gap between the application and network worlds of standards. It is also the one new protocol where enterprises could define an interface between the parallel enterprise fiefdoms: the application and network groups.

The session protocol has been a long time coming, but we really need it now. Those old host-based control freaks had something to tell us. Time to listen to the old guidance and finish the ISO model in the Internet way. The only question is who will take on this standards work?

Vendor Strategies

This market got a big boost from recent acquisitions by Cisco and Juniper. Both purchased what we call an application-centric and a network-centric solution (see BCR, April 2005, pp. 37-43). There are at least two questions: "How will they integrate these products into a cohesive enterprise solution?" and "Will they see the need for the session protocol as an element of that solution?"

Other leading vendors have a similar multi-pronged approach to the problem: Blue Coat, Expand Networks, Packeteer, and Swan Labs. These smaller, more agile vendors may lead the charge toward a comprehensive solution, and are more likely to start defining a session strategy.

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. It will be important to watch the leading players to see if they develop comprehensive long-term solutions.

Companies mentioned
Blue Coat (www.bluecoat.com)
Cisco (www.cisco.com)
Expand Networks (www.expand.com)
Juniper (www.juniper.net)
Packeteer (www.packeteer.com)
Swan Labs (www.swanlabs.com)

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.