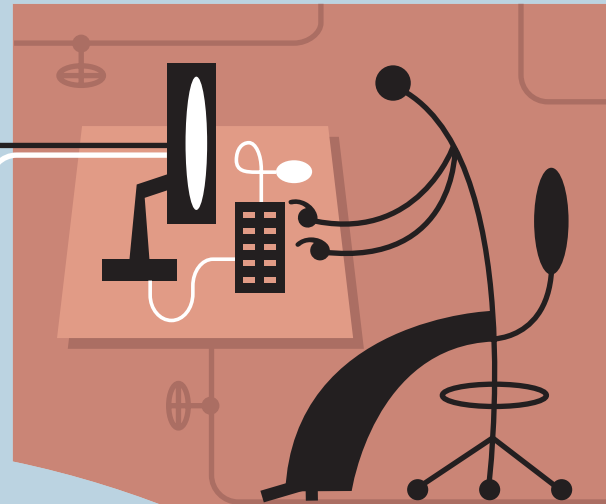
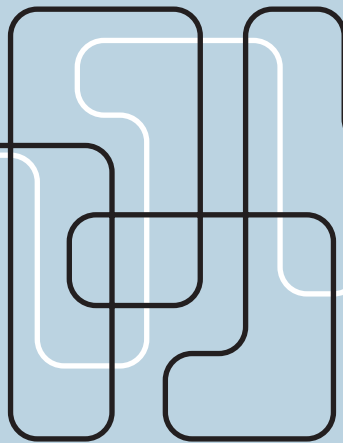




► Applications



Network ◀

When it comes to the divide between application developers and network architects, it's time to

Stop the madness!

BY BILL LABERIS

Illustration by Scott Roberts

The application team is writing a new application. They put it through a standard set of tests and prototyping; then they throw it over the wall to the network team to deploy. The apps team fully expects this baby to work flawlessly. (After all, it worked in the testing phase!)

The application goes live, and it doesn't work nearly as well as expected. What happens next? The network team is told, "Fix the problem. It must be the network." The network team either buys a ton of new bandwidth to "fix" performance, or they scramble to deploy application delivery controllers and WAN optimization tools—none of which have been budgeted for. The business manager who ordered the new application in the first place screams when she sees the increased costs, which now wipe out a second business-critical application she had planned on deploying. Fingers are pointed. Tempers flare.

Does this sound familiar?

To quote one of the most classic of movie lines: *What we've got here is a failure to communicate.* The historical divide between application developers and network architects is now starting to get in the way

CAN WE TALK?

If there is one thing all the experts agree upon, it is the importance of simply getting all the interested and involved parties to sit down in the same room prior to the first line of application code being written and hash out the network implications of the planned deployment. In other words, the resolution to the problem of bridging the divide between applications and networks is about internal politics and processes as well as technology. And when you consider that network and application teams generally have no common reporting structure until you reach the CIO level, it's really up to the leaders in each organization to grab the reins and get both groups around the same conference table. Here are a few questions to get the discussion moving in that initial meeting.

- Were there problems with recent application deployments, particularly if these deployments were undertaken in an SOA environment?
- Were there chronic problems with unacceptable latency or actual server downtime?
- Is the application team regularly finding itself changing code or adding new security features and functions due to shifting application requirements?
- With the planned applications, what is known about peak loads and when they will occur?
- How many new transactions per day are anticipated, and where are the people who will be running those transactions?
- What database is being used, and how should the data be structured?
- Are there business processes we would like to automate?
- Can the application team articulate its modeling and prototyping procedures clearly to the network team?
- Do all team members understand the wisdom and potential of embedding more intelligence into specialized network components early in the process?

of real progress in aligning IT with business. As the complexity of the IT environment increases with the dawning of the SOA era and with the rising tide of regulation, things can quickly go from bad to really bad to chaotic.

It's time to stop the madness and get the application and network teams on the same page, long before deployment time. But how? What are the impediments to closing the communication gap? What are the payoffs? How much of the solution is technology, versus how much is politics and process?

More important than ever

Historically, the fact that they always worked in relative isolation from each other hasn't been a big problem for application and network teams. Application rollouts were more local than global, so local testing could identify many problems. In that pre-SOA timeframe, application developers well understood the relatively simple nature of data, and that made guesstimates of transactions and network impact relatively straightforward.

IT in general did not need to be as flexible and adroit in past years as it does today and certainly will in the future, so problems could be worked out over time without severely affecting the business.

Today's reality is quite different. SOA-enabled applications, with their bulky, message-based XML traffic, result in very heavy network loads, often unanticipated until the application is rolled out. A request or command that took two round-trips to the server in the pre-SOA world now takes 20 trips. As Joe Skorupa, a research vice president at Gartner Inc. puts it, "Developers who write protocols often don't think about latency or bandwidth, because they assume zero latency and infinite bandwidth. There is little sensitivity to these key network issues, because in the past there was little need for concern."

Lastly, the globalization of business means that applications run on a variety of networks and devices too numerous to anticipate by application developers.

Most testing doesn't work

"But we tested the application and it worked," says the application architect. But often that testing consisted of a local rollout with, say, 200 users—for an application designed to reach a global network of users. Or the testing was carried out over local connections directly to servers at gigabit speeds.

In most cases, the fault lines with testing-gone-bad can be traced to the application not having been tested across a true, emulated network. Assuming the application is truly business-critical, that testing should involve a broad range of IT experts, including infrastructure specialists, storage gurus, security folks, and even consultants who really understand the interplay of applications and the rising tide of regulatory compliance. It is crucial to budget for all this additional planning, too.

It's the people

Oddly, and perhaps rather refreshingly, the impediments to bridging the historical divide between two highly technical IT groups are generally not the result of technology (see related story, page 11). "It's all about knowledge of one another's domains and about education, breaking down walls that have been up forever," notes Dan Matte, senior vice president of marketing at F5. "The technology is largely in place and can be as simple as a customer writing iRules on BIG-IP to check for problems before they spill out."

Joe Pruitt, senior strategic architect at F5, agrees: "Developers just weren't trained to think about deployment as much as they think about features and deliverables and schedules. They don't want to be called at 2 a.m. because the network has been brought to a crawl by the application. On the other side, the network team needs to better understand how they can impact application logic. It's amazing what can happen if they just sit down before any real shooting begins and talk." *

F5 CAN HELP

Joe Skorupa, research vice president at Gartner, says embedding greater intelligence into specialized network components can improve application performance by an order of magnitude or greater.

Skorupa adds that letting network intelligence offload processes that "have no business running on a Web or application server" can help application developers get a more realistic picture of how their applications will perform during the development process. He cites examples of companies that wasted significant development time and considerable money

building processing capacity into servers when a properly equipped network could have done the job more efficiently.

F5, of course, offers numerous tools, devices, and solutions for building the kinds of application-fluent networks increasingly required for today's SOA-enabled applications. For example, using F5's iRules, many companies are learning to adapt far more quickly and painlessly to changing application requirements.

But in addition, F5's website is becoming a virtual meeting place for application developers and network professionals from a range of organizations. Leveraging

F5 DevCentral (<http://devcentral.f5.com>) has helped these two communities find common ground.

"Reading the exchanges among application and network people, I'll see comments like, 'I never realized we could do that kind of thing, but now I see plenty of people trying it out,'" says Joe Pruitt, senior strategic architect at F5. "With podcasts, blogging, and other resources, DevCentral is becoming a hub for uniting these two groups."

In addition, F5 recently opened its Seattle-based Technology Center (see related story, p. 16). Here customers can send both their application

and network architects to help prove out their plans and concepts by running applications in their own simulated environments and then using F5 technology to offload processing overhead.

And when it comes to using technology to bridge the network-applications divide, even F5 turns to F5, Pruitt says. "In a recent case, instead of redeveloping all the code in our back-end application server, we wrote a three-line iRule, put it on BIG-IP, and solved that issue. It could have taken us literally days to recode, but instead we were able to deploy the changes into the network in about eight minutes."