Chapter

2

What We Found in Our Travels

Since our first book was published in 1994, we've worked with companies all over the world, presenting at large functions, and teaching classes on the subject of how to effectively implement and manage client/server infrastructures. We provided insight on how to deal with the people and process issues. The reception and feedback were very positive. Since 1994 we've published four additional books on this same subject. Sales figures were excellent so we assumed the interest was still there, but why aren't all these infrastructure-related problems with client/server computing being addressed?

In our travels we learned a heck of a lot in visiting with more than 500 companies (probably closer to 1,000). The most apparent issue is how inefficiently client/server environments are being run. But that's putting it mildly. It's been getting worse year after year, and what's so disheartening is that there is very little to look forward to in the next several years.

These corporations are having a terrible time attaining RAS. No wonder metrics weren't being kept on server availability. How embarrassing it is to show that your servers were up 70 to 80 percent of the time. The customers were complaining about the hardware and their vendor without stepping back to look and see if their house (infrastructure) was in order.

In this section we document the problems from data gathered from more than 40 case studies and share the bad news with you.

What's Wrong with Client/Server?

The problems are enormous! Client/server computing is flying by the seat of its pants. There's no planning. There are no metrics. Roles and responsibilities are not clearly defined. There are very few processes that are implemented, maintained, and monitored. Communication within Information Technology and with IT's customers is worse than ever before. It's a real mess out there. But this is only the beginning.

It's no shock to IT professionals that there are problems with client/server computing but what's eye opening are the number and severity of the problems. We traveled the world in search of a well run client/server-computing environment with RAS to benchmark. And we searched and searched. We searched for over two years and we came up dry. What in the world is going on here? We needed to find the answers.

Our data was compiled from performing one- to two-day IT infrastructure assessments. We ended up performing more than 40 of these assessments with major corporations around the world. The results were disheartening and difficult to fathom.

The Problems

Table 2–1 points out the most common problems and their overall impact on the infrastructure.

Table 2–1   Properties of Fact and Dimension

Problem	Impact
Many IT shops are organizing to focus on particular technologies (i.e. Mainframe, AS400, NT, Unix, Novel, etc.).	● Poor communication among groups. ● Duplication of System Management efforts. ● Poor morale. ● Lack of enterprise-wide solutions. ● Huge walls between the groups. ● Limited technical resources.
Some organizations structure to focus on high-visibility projects and have a separate structure to focus on production support.	● Difficult to turn over projects from development to support. ● Poor morale. ● Technicians would prefer to work on new projects rather than be labeled as full-time, production-support personnel

Table 2–1   Properties of Fact and Dimension

Problem	Impact
The Architecture function has proven to be ineffective for designing infrastructures. The CIO might think otherwise because one of the architect's functions is to help design the proper infrastructure. It may be their function but we have yet to see this work effectively.	● Infrastructure development lags further behind the needs of the customers and IT.
Lack of a production control function (production Q/A, second-level system administration, process ownership, etc.).	● There's only one level of support for System Administration functions — all problems go directly to senior technical staff. ● Lack of junior system administrator skill development. ● systems management tools not fully implemented, customized, and maintained. Not enough time to do the job right. ● Senior technical staff cannot properly plan and design the infrastructure because they're too busy firefighting problems.
Duplication of system management efforts. In many instances staff is only looking at point solutions to address the issues to support the technology for which they are responsible.	● Wasted technical resource cycles. ● Higher costs to IT. ● Lack of enterprise-wide solutions. ● Fingerpointing among groups.
Lack of skilled resources. Most IT shops we visited are not taking the time to breed technical expertise within the organization. They're only looking at external recruitment efforts.	● The competition is fierce in the marketplace. Most of the companies we visited with are putting all their energy into external recruiting. This leaves a big void—they need to start breeding within the organization as well as continue to look at external resources.
Retaining senior technical staff is difficult. In all the environments we visited the technical staff is in a constant firefighting/reactive mode.	● Frustration of staff because of chaotic state of the infrastructure. ● Career development is limited to solving day-to-day problems. ● Burnout is imminent.
Lack of junior technical staff career development. The organization doesn't promote the proper career path.	● Management can only recruit technical staff from limited external sources. ● Lack of skilled technical resources.

Table 2–1 Properties of Fact and Dimension

Problem	Impact
Job functions overlapping. Roles and responsibilities are not clearly defined.	● Duplication of efforts. ● Confusion in the ranks for problem resolution, especially for the Help Desk staff as they're tasked to resolve problems as quickly as possible. ● Problem management is difficult to define and administer. ● Staff is responsible for many things but very few own any of it. ● Production support confusion reigns throughout IT.
Communication is extremely poor.	● Wasted efforts. ● Inefficient use of resources. ● Projects take more time, resources, and money to implement. ● Service levels difficult to maintain. ● Users frustrated with IT.
Problem management is ineffective.	● High availability becomes an unattainable goal. ● Lack of problem ownership. ● People are tied up in priorities—they don't have the time to document the problems/issues. ● Lack of follow-up. ● Problems get lost—no tracking. ● There's a lack of information. ● Lack of root cause analysis. ● Lack of closed-loop feedback. ● Level 2 analysts not putting in detailed description of how they resolved the problem. ● Many of the groups provide no feedback on problems being worked.
Multiple Help Desks.	●Confusion for end-users. ●Duplication of efforts. ●Multiple owners — or no ownership — of a very critical problem management process. ●Higher costs.
Lack of effective processes.	RAS unattainable.

Table 2–1   Properties of Fact and Dimension

Problem	Impact
Help Desk improperly structured within the organization.	● Lack of authority.
LAN support is split among multiple organizations.	● Confusion for end-users. ● Duplication of efforts. ● Higher costs to IT.
Database administration not centralized; in many companies it's organized under applications development, for others in operations support, and for others it's split between the two.	● Causes poor communication. ● Problem resolution not as effective. ● Duplication of efforts. ● Not equal emphasis on analysis and administration
Some IT shops are splitting the infrastructure group between infrastructure development and production support.	● Poor communication. ● Poor morale. ● Duplication of efforts. ● Turf battles.
System management tools are not fully implemented.	● Manual intervention. ● Wasted costs. ● Occasional glitches. ● Wasted technical resources.
Lack of a tape librarian function.	● Integrity compromised. ● Minimum disaster recovery requirements are nonexistent.
Global coordination is not effective.	● Poor communication. ● Frustration. ● Duplication of efforts.
Lack of an effective change control process.	● RAS compromised.
Lack of a client/server application production acceptance process.	● No production QA function. ● Poor communication between IT and its users. ● Production support and applications development are not communicating early on in the application deployment cycle.
Lack of metrics.	● Cannot effectively manage the infrastructure unless you know the numbers.

Table 2–1 Properties of Fact and Dimension

Problem	Impact
Lack of an effective curriculum to effectively transition the legacy staff.	● Limited technical resources. ● Poor morale. ● Wasted IT costs.
Lack of a strategic process to market and sell IT services. IT needs to communicate its services with its customers.	● Poor relations with customers. ● Users don't know where to turn for services
Lack of service levels.	● Expectations not properly documented between end-users and IT.
Lack of a process to benchmark services.	● Management can continue threatening to outsource IT.
Not measuring customer satisfaction	● Disastrous
Lack of data center mentoring.	● Moving forward to effectively support mission critical client/server applications regardless of the platform and paradigm is extremely difficult. It is critical to take the best practices from the legacy environment and the most important methodologies from open systems to come up with the best of both worlds.
Complexity in the organization structure.	●Ineffective use of resources. ●Poor communication. ●Duplication of efforts.
Lack of standards and lack of adherence to standards throughout the enterprise	●Duplication of efforts. ●Poor communication. ●Higher costs. ●Wasted technical resources.

There are so many problems with today's infrastructure. The top problems (in no particular order except the first) in today's client/server environments, based on the 40 assessments are:

1. The organization structure
2. Lack of an enterprise-wide change management process
3. Lack of an effective problem management process
4. Lack of a production acceptance process

5. Lack of metrics

6. Lack of a proper curriculum to transition/mentor staff

7. Communication is worse than ever before

8. Not fully implementing system management tools

9. Lack of senior technical resources

10. Lack of a process to market/sell and benchmark IT services

11. Lack of service levels between operational support and applications development and also between IT and its customers

12. Recruiting/retaining technical resources

These aren't surprising to IT professionals. But what is astonishing is that the No. 1 problem is the organization structure for each and every company we visited. This is incredible!

After the first half dozen assessments, we realized that client/server computing was making a mockery of the IT profession because many of the problems pointed back to the organization. Each one of these assessments highlighted major problems with the way the organization was structured. How can that be? Structuring the organization to support a new paradigm couldn't be that difficult. It isn't, but the biggest problem is that IT executives know that client/server computing is so much more complex. So they try to overcompensate by focusing on particular technologies or high-visibility projects.

Reorganizations are occurring at a torrid pace, like never before. Most of the companies we visited reorganized at a minimum of every six months. In the legacy days, once a year was a big-news event. But we can't blame all these intelligent executives. Something needs to be done to introduce RAS into these chaotic networked computing environments.

So why is network computing so difficult to get our arms around? Very few understand where the problems really are. With so much talk of technology these days, many IT professionals look at technology as their savior, the more the better to solve these problems. They gobble it up and plop it down. It seems logical. So why isn't it working? Why can't companies attain a high-level of RAS as we have on the mainframe? The technology everyone's gobbling up is definitely mature enough to manage a mission critical production environment.

Legacy-minded critics blame client/server computing. This is ridiculous. For decades we focused all energy on people and process issues to

have an environment that is reliable, available, and serviceable. RAS starts with the organization.

This is not rocket science. Yet, why is it so difficult to structure the organization to accommodate RAS when we have all these highly educated and experienced IT professionals? Executives have used every excuse in the book to reorganize.

The Excuses

- The business is changing.
- We have some hot new technology requiring resources.
- Our company is downsizing; we need to restructure.
- We have to separate the legacy environment from the other stuff because we want to sustain RAS.

The real issue is client/server computing. It is like nothing IT veterans or, for that matter, younger executives have ever come across. Yet they're not admitting it. How do you structure for something that has no boundaries or clear demarcations? We'll show you throughout this book.

What made us so smart? Why did we find the answers? Two reasons. First, our backgrounds helped. We were reared in a legacy mission-critical data center environment. We were also chartered with transitioning large legacy computing environments to client/server. Second, we have no biases. There is such an ugly perception that managers of legacy computing environments are bureaucratic, maintain costly environments, and are behind the times, etc. Sure, a lot of that is true, but what about RAS? What about when you came to work and the systems were always up and running? That's where mission-critical, high-availability evolved. All we ask people to do as they read the book is to look at what worked and what didn't work in the legacy environment to effectively move forward.

Applications Development Is Not an Issue

Why did we pick on the infrastructure group and say nothing about applications development? Surely applications development can't be

perfect. It's not, but the issues are small compared to the infrastructure group. In the late eighties and throughout the nineties the focus was to develop and deploy quickly. The infrastructure took the backseat. In the seventies the Infrastructure drove the car and owned the back seat—applications development was somewhere in the trunk. Times have changed. You need both groups playing together to be successful. Without the proper infrastructure there cannot be RAS when deploying new systems, and when the systems cannot maintain high availability, all of IT gets a black eye.

The Impact

There is, however, an impact from the infrastructure group that hits applications development right where it counts — rolling out new production-quality systems. When deploying new systems there needs to be a process that ensures quality assurance (QA), that promotes and instills effective communication practices between applications development and production support. There needs to be a checklist to engage key personnel from the beginning. This same checklist needs to ensure adherence to disciplines (i.e., change control). There is such a process, which we refer to as the Client/Server Production Acceptance (CSPA). Please read Chapter 7 in our book *Building the New Enterprise* or Chapter 9 in our book *Managing the New Enterprise*.

The two groups need to work together when deploying new systems. applications development needs production support, and production support has a responsibility to service applications development.

Processes Are Not First

Organizing to support RAS should be first. Structure the organization properly to support a mission-critical environment. Don't reorganize because you're implementing a new process. Many shops do this, but it's wrong. Once you restructure to support RAS, the organization will have clearly-defined process ownership. Processes need one owner for design, implementation, and maintenance.