Creating Unified IT Monitoring and Management in Your Environment

Don Jones

Introduction to Realtime Publishers

by Don Jones, Series Editor

For several years now, Realtime has produced dozens and dozens of high-quality books that just happen to be delivered in electronic format—at no cost to you, the reader. We've made this unique publishing model work through the generous support and cooperation of our sponsors, who agree to bear each book's production expenses for the benefit of our readers.

Although we've always offered our publications to you for free, don't think for a moment that quality is anything less than our top priority. My job is to make sure that our books are as good as—and in most cases better than—any printed book that would cost you $40 or more. Our electronic publishing model offers several advantages over printed books: You receive chapters literally as fast as our authors produce them (hence the "realtime" aspect of our model), and we can update chapters to reflect the latest changes in technology.

I want to point out that our books are by no means paid advertisements or white papers. We're an independent publishing company, and an important aspect of my job is to make sure that our authors are free to voice their expertise and opinions without reservation or restriction. We maintain complete editorial control of our publications, and I'm proud that we've produced so many quality books over the past years.

I want to extend an invitation to visit us at http://nexus.realtimepublishers.com, especially if you've received this publication from a friend or colleague. We have a wide variety of additional books on a range of topics, and you're sure to find something that's of interest to you—and it won't cost you a thing. We hope you'll continue to come to Realtime for your educational needs far into the future.

Until then, enjoy.

Don Jones

Copyright Statement

Realtime
publishers

Chapter 1: Managing Your IT Environment: Four Things You're Doing Wrong

At the very start of the IT industry, "monitoring" meant having a guy wander around inside the mainframe looking for burnt-out vacuum tubes. There wasn't really a way to locate the tubes that were working a bit harder than they were designed for, so monitoring—such as it was—was an entirely reactive affair.

In those days, the "Help desk" was probably that same guy answering the phone when one of the other dozen or so "computer people" needed a hand feeding punch cards into a hopper, tracking down a burnt-out tube, and so on. The concepts of tickets, knowledge bases, service level agreements (SLAs), and so forth hadn't yet been invented.

IT management has certainly evolved since those days, but it unfortunately hasn't evolved as much as it could or should have. Our tools have definitely become more complex and more mature, but the way in which we *use* those tools—our IT management processes— are in some ways still stuck in the days of reactive tube-changing.

Some of the philosophies that underpin many organizations' IT management practices are really becoming a detriment to the organizations that IT is meant to support. The discussion in this chapter will revolve around several core themes, which will continue to drive the subsequent chapters in this book. The goal will be to help change your thinking about how IT management—particularly monitoring—should work, what value it should provide to your organization, and how you should go about building a better-managed IT environment.

IT Management: How We Got to Where We Are Today

In the earliest days of IT, we dealt with fairly straightforward systems. Even simplistic, by today's standards. The IT team often consisted of people who could fix any of the problems that arose, simply because there weren't all that many "moving parts." It's as if IT was a car: A machine capable of complexity and of doing many different things, but perfectly comprehendible, in its entirety, by a single human being.

As we started to evolve that IT car into a space shuttle, we gradually needed to allow for specialization. Individual systems became so complex in and of themselves that we needed domain-specific experts to be able to monitor, maintain, and manage each system. Messaging systems. Databases. Infrastructure components. Directory services. The vendors who produced these systems, along with third parties, developed tools to help our experts monitor and manage each system. That's really where things went wrong. It seemed perfectly sensible at the time, and indeed there was probably no other way to have done things, but that establishment of domain-specific silos—each with their own tools, their own procedures, and their own expertise—was the seed for what would become a towering problem inside many IT shops.

Fast forward to today, when our systems are vastly more complex, vastly interconnected, and increasingly not even hosted within our own data centers. When a user encounters a problem, they obviously can't tell us *which* of our many complex systems is at fault. They simply tell us what they observe and experience about the problem, which may be the aggregate result of several systems' interactions and interdependencies. Our users see a holistic environment: "IT." That doesn't correspond well to what we see on the back end: databases, servers, directories, files, networks, and more. As a result, we often spend a lot of time trying to track down the root cause of problems. Worse, we often don't even see the problems coming, because the problems only exist when you look at the end result of the entire environment rather than at individual subsystems. Users feel completely disconnected from the process, shielded from IT by a sometimes-helpful-sometimes-not "Help desk." IT management has a difficult time wrapping their heads around things like performance, availability, and so on, simply because they're forced to use metrics that are specific to each system on the network rather than look at the environment as a whole.

The way we've built out our IT organizations has led to very specific business-level issues, which have become common concerns and complaints throughout the world:

- IT has difficulty defining and meeting business-level SLAs. "The messaging server will be up 99% of the time" isn't a business-level SLA; it's a technical one. "Email will flow between internal and external users 99% of the time" is a business-level SLA, but it can be difficult to measure because that statement involves significantly more systems than just the email server.

- IT has difficulty proactively predicting problems based on system health, and remains largely reactive to problems.

- When problems occur, IT often spends far too much time pinpointing the root cause of the problem.

- IT's concept of performance and system health is driven by *systems*—database servers, directory services, network devices, and so forth—rather than by how users and the organization as a whole are experiencing the *services* delivered by those systems.

- IT has a tough time rapidly adopting new technologies that can benefit the business. Oxymoronically, IT is often the part of the organization most opposed to change, because change is usually the trigger for problems. Broken systems don't help anyone, but an inability to quickly incorporate changes can also be a detriment to the organization's competitiveness and flexibility.

- IT has a *really* tough time adopting new technologies that are significantly outside the team's experience or physical reach—most specifically the bevy of outsourced offerings commonly grouped under the term "cloud computing." These technologies and approaches to technology are so different from what's come before that IT doesn't feel confident that they can monitor and manage these new systems. Thus, they resist implementing these types of systems for fear that doing so will simply damage the organization.

- Even with modern self-service Help desk systems, users feel incredibly powerless and out of touch when it comes to IT.

All of these business-level problems are the direct result of how we've always managed IT. Our processes for monitoring and managing IT basically have four core problems. Not every organization has every single one of these, of course, and most organizations are at least aware of some of these and work hard to correct them. Ultimately, however, organizations need to ensure that all four of these core problems are addressed. Doing so will immediately begin to resolve the business-level issues I've outlined.

Problem 1: You're Managing IT in Silos

Figures 1.1, 1.2, and 1.3 illustrate one of the fundamental problems in IT monitoring and management today.
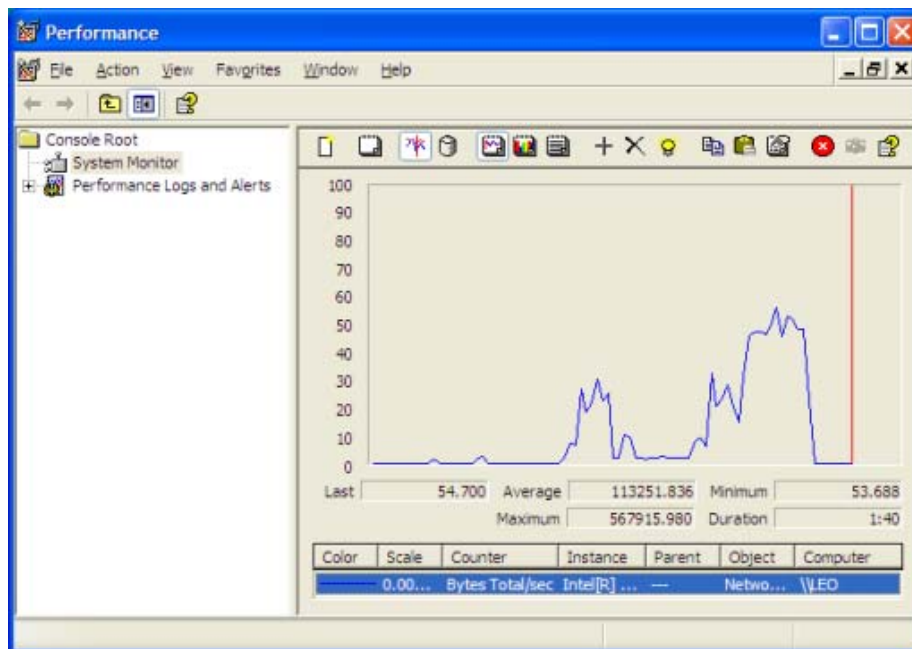
Figure 1.1: Windows Performance Monitor.

Figure 1.2: SQL Server Performance.
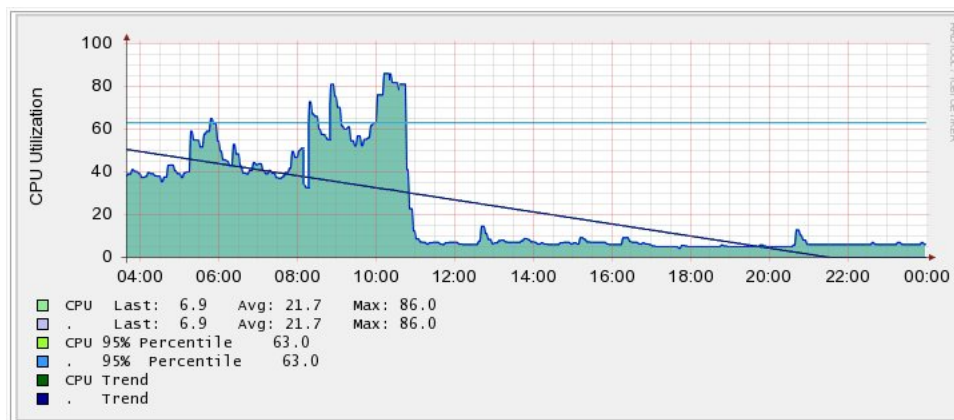
Figure 1.3: Router Performance.

These figures each illustrate a different performance chart for various components of an IT system. Each of these images was produced using a tool that is more or less specialized for the exact thing that was being monitored. The tool that produced the router performance chart, for example, can't produce the same chart for a database server or even for a router that's located on someone else's network.

This is such a core, fundamental problem that many IT experts can't even recognize that it *is* a problem. Using these domain-specific tools is such an integrated and seemingly natural part of how IT works that many of us simply can't imagine a different way. But we *need* to move past using these domain-specific tools as our first line of defense when it comes to monitoring and troubleshooting.

Why?

One major reason is that these tools keep us all from being on the same page. IT experts can't even have meaningful cross-discipline discussions when these tools become involved. "I'm looking at the database server, and the performance is at more than 200 TPMs," one expert says. "Well, that must be a problem because the router is running well over 10,000 PPMs." Those two experts don't even have a common language for performance because they're locked into the domain-specific, deeply-technical aspects of the technologies they manage.

Domain-specific tools also encourage what is probably the worst single practice in all of IT: looking at systems in isolation. The database guy doesn't have the slightest idea what makes a router tick, what constitutes good or bad performance in a messaging server, or what to look for to see if the directory services infrastructure is running smoothly. So the database guy puts on a set of blinders and just looks at his database servers. But those servers don't exist in a vacuum; they're impacted by, and they in turn impact, many other systems. Everything works *together,* but we can't *see* that using domain-specific tools.

We *have* to permanently remove the walls between our technical disciplines, breaking down the silos and getting everyone to work as a single team. In large part, that means we're going to have to adopt new tools that *enable* IT silos to work as a team, putting the information everyone needs into a common context. Sure, domain-specific tools will always have their place, but they can't be our first line of information.

> **Case Study**
>
> Jerry works for a typical IT department in a midsize company. His specialty is Windows server administration, and his team includes specialists for Web applications, Microsoft SQL Server and Oracle, VMware vSphere, and for the network infrastructure. The company outsources certain enterprise functionality, including their Customer Relationship Management (CRM) and email.
>
> Recently, a problem occurred that caused the company's main Web site to stop sending customer order confirmation emails. Jerry was initially called to solve the problem, on the assumption that it was with the company's outsourced messaging solution. Jerry discovered, however, that user email was flowing normally. He passed the problem to the Web specialist, who confirmed that the Web site was working properly but that emails sent by it were being rejected. Jerry filed a ticket with the messaging hosting company, who responded that their systems were in working order and that he should check the passwords that the Web servers were using.

> After more than a day of back-and-forth with the hosting company and various experts, the problem was traced to the company's firewall. It had recently been upgraded to a new version, and that version was now blocking outgoing message traffic from the company's perimeter network, which is where the Web servers were located. The network infrastructure specialist was called in to reconfigure the firewall, and the problem was solved.

This narrative precisely demonstrates the problem: By managing our IT teams as domain-specific silos, we significantly hinder their ability to work together to solve problems. The fact that IT experts require domain-specific tools shouldn't be a barrier to breaking down those silos and getting our team to work more efficiently together. This becomes especially important when pieces of the infrastructure are outsourced; those hosting companies are an unbreakable silo, as they're not responsible for any systems other than the ones they provide to us. However, the dependencies that our systems and processes have on their systems means our own team still has to be able to monitor and troubleshoot those outsourced systems *as if* they were located right in the data center.

Problem 2: You Aren't Connecting Your Users, Service Desk, and IT Management

Communication is a key component of making any team work; and the "team" that is your organization is no exception. In the case of IT, we typically use Help desk systems as our means of enabling communications—but that isn't always sufficient. Help desk systems are almost always built around the concept of reacting to problems, then managing that reaction; they're almost by definition not *proactive.*

For example, how do you tell your users that a given system will have degraded performance or will be offline for some period of time? Probably through email, which creates a couple of problems:

- Important messages tend to get lost in the glut of email that users deal with daily
- Users who don't get the message tend to go the "Help desk route," which doesn't include a means of intercepting their mental process and letting them know that the "problem" was planned for.

Realtime
publishers

Most IT teams *do* know the things that need to be communicated throughout the organization, for example:

- SLAs

- The current status of SLAs—whether they're being met

- Planned outages and degraded service

- Average response times for specific services

- Known issues that are being worked on

What most IT teams have a problem with is communicating these items consistently across the entire organization. Some organizations rely on email, which as I've already pointed out can be inefficient and not consistently effective. Some organizations will use an intranet Web site, such as a SharePoint portal, to post notices—but these sites aren't directly integrated with the Help desk, making it an extra step to keep them updated and requiring users to remember to check them.

Case Study

Tom works as an inside salesperson for a midsize manufacturing company. Recently, the application that Tom uses to track prospects and create new orders started responding very slowly, and over the course of the day, stopped working completely.

Tom's initial action was to call his company's IT Help desk. The Help desk technician sounded harried and frustrated, and told Tom, "We know, we're working on it," and hung up. Tom had no expectation when the system might return to normal, and was afraid to bother the Help desk by calling back for more details.

Over the course of that day, the Help desk logged calls from nearly every salesperson, each of whom called on their own to find out what was going on. Eventually, the Help desk simply stopped logging the calls, telling everyone that, "A ticket is already open," and disconnecting the call.

Someone on the IT management team eventually sent out an email explaining that a server had failed and that the application wasn't expected to be online until the next morning. Tom wished he had known earlier; although he'd originally planned to make sales calls all day, if he'd known that the application would be down for that long, he could have switched to other activities for the day or even just taken the day off.

Management communications are equally important, and equally challenging. Providing frank numbers on service levels, response times, outages, and so forth is crucial in order for management to make better decisions about IT—but that information can often be difficult to come by.

Problem 3: You're Measuring the Wrong Things

This problem is very likely at the heart of everything IT is *not* doing to help better align technology with business needs. The following case study outlines the scenario.

> **Case Study**
>
> Shelly works in the Accounting department for her company. Recently, while trying to close the books for her company, the accounting application began to react very slowly. She called her company's IT Help desk to report the problem.
>
> The Help desk technician listened to her then said that, "Everything on that server looks fine right now. I'll open a ticket and ask someone to look at it, but since we are currently within our service level agreement for response times, it will be a low-priority ticket."
>
> Shelly continued to struggle with the slowly-responding application. Eventually, someone was dispatched to her desktop. She demonstrated that every other application was responding normally. She pointed out that other people in her department were having similar problems with the application. The technician made her close all of her applications and then restarted her computer, to no effect. He shrugged, entered some notes into his smartphone, and left.
>
> By the next morning, the application's response times were better, but they were far from normal. Shelly continued to call the Help desk for updates on her ticket's status, but it seemed as if the IT team had given up on trying to fix the problem—and refused to even admit that there *was* a problem.

This kind of scenario unfortunately happens all too often in many organizations. It exactly illustrates what happens when several problems are happening at once: IT is operating as a set of individual silos rather than as a team, and each silo has its own definition for words like "slow." A root issue here is that *everyone is measuring the wrong thing.* Figure 1.4 shows how the average IT team sees a multi-component, distributed application.
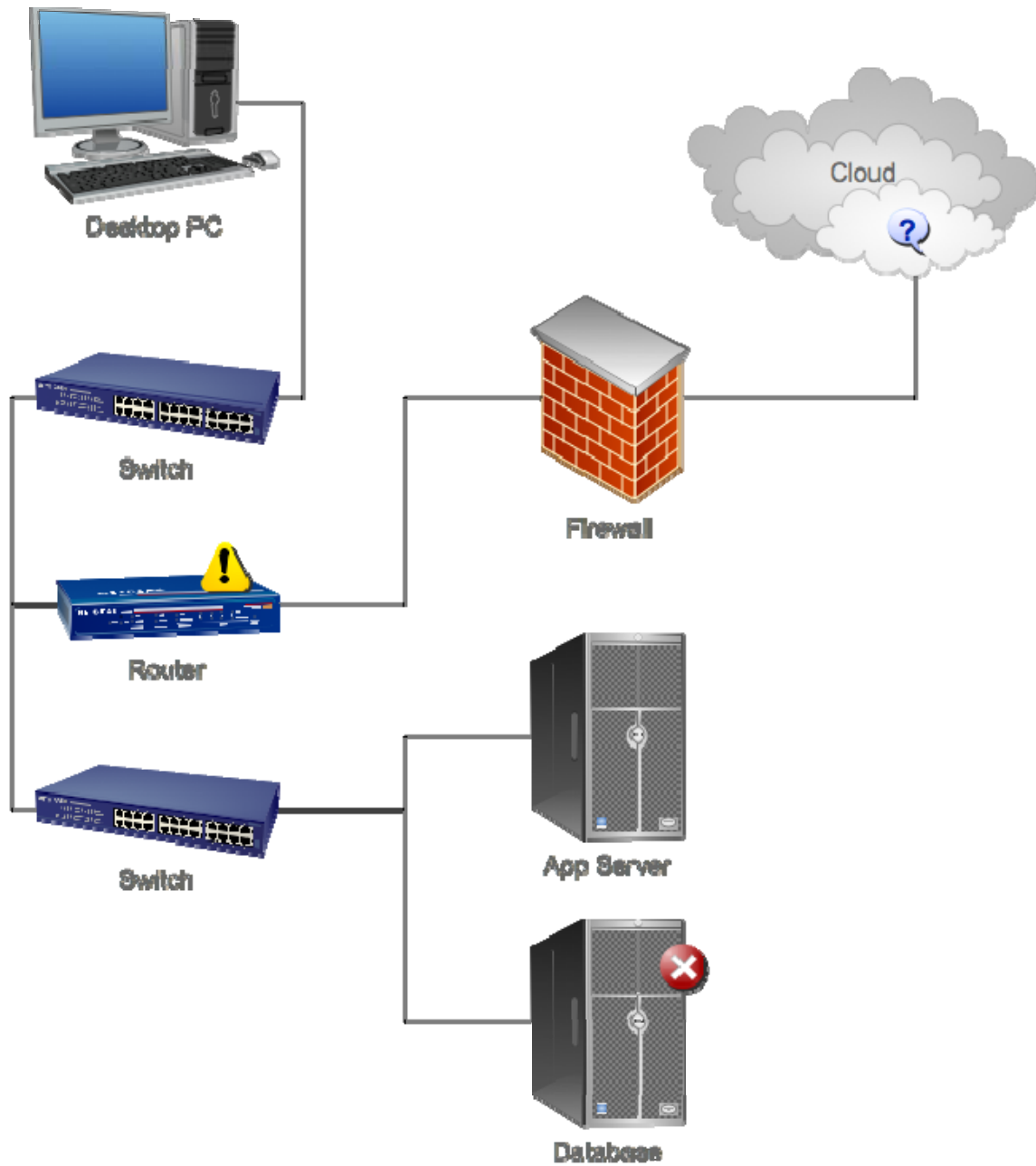
Figure 1.4: IT perspective of a distributed application.

They see the components. Domain experts measure the performance of each component using technical metrics, such as processor utilization, response time, and so forth. When a component's performance exceeds certain predefined thresholds, someone in IT pays attention. Figure 1.5, however, shows how a user sees this same application.

Figure 1.5: User's perspective of a distributed application.

The user doesn't—often *can't*—see any of the components. They simply see an application, and either it's responding the way they expect, or it isn't. It doesn't matter a bit to the user if every single constituent component is running at an "acceptable level of processor utilization"—whatever *that* means. They simply care whether the application is working. This creates a major disconnect between the user population and IT, as Figure 1.6 illustrates.

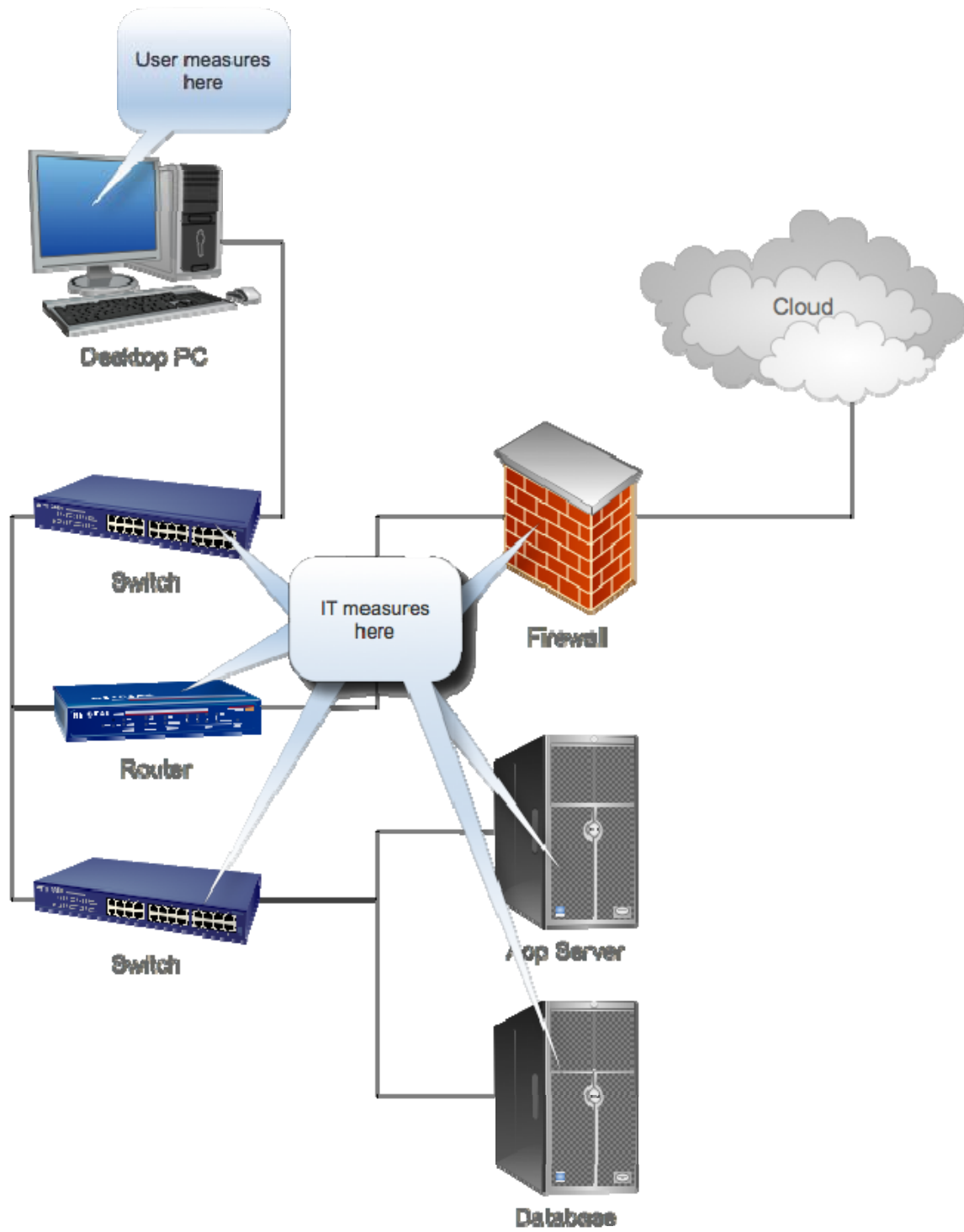**Figure 1.6: IT vs. user measurements of performance.**

Users and IT measure very different things. An IT-centric SLA might specify a given response time for queries sent to a database server; that often has little to do with whether an application is seen as "slow" by users. Worse, as we start to migrate services and components to "the cloud," we lose much of our ability to measure those components' performance the way we do for things that are in our own data center. The result? Nobody can agree on what an SLA should say.

This all has to change. We have to start measuring things more from a user perspective. The performance of individual components is important, but only as they contribute to the total experience that a user perceives. We need to define SLAs that put everyone—users and IT—on the same page, then manage to those SLAs using tools that enable us to do so.

Some organizations will tell you that they're moving, or have moved, to a service-based IT offering. What that generally means in broad terms is that the organization is seeking to provide IT as a set of services to the organization's various departments and users. In many instances, however, those "service-oriented" organizations are still focused on components and devices, which isn't a service-oriented approach at all. When your phone line goes down, you don't call the phone company (on your cell phone, probably) and start asking questions about switches and trunk lines—you ask when your dial tone will be back. The back-end infrastructure is meaningless to the user. You don't ask for a service credit based on how long a particular phone company office will be offline, you ask for that credit based on how long you went without a dial tone. That's the model IT needs to move toward.

Problem 4: You're Losing Knowledge

The last problematic practice we'll look at is the issue of lost institutional knowledge. This problem is a purely human one, and frankly it's going to be difficult to address. Here's a quick scenario to set the scene.

> **Case Study**
>
> Aaron works for his company's IT department. He's been with the company for 3 years and is responsible for several of the company's systems and infrastructure components. One Tuesday, Aaron is contacted by his company's IT Help desk. "We're assigning you a ticket about the Oracle system," he's told. "Once every couple of months it starts acting really weird, and someone has to fix it."
>
> "I'm not the Oracle guy," Aaron says. "That's Jill."
>
> "Yeah, but Jill's out on vacation for 2 weeks. So you'll have to fix it."
>
> "I've no idea what to do!"
>
> "Well, figure something out. The CEO gets upset when this takes too long to fix."

Unfortunately, too much knowledge gets wrapped up in the heads of specific individuals. In fact, it's a sad truth that many organizations "deal" with this problem by simply discouraging IT team members to take lengthy vacations, and often resist other activities that would put them out of touch—such as sending them to conferences and classes to continue their education and to learn new skills.

More than a few organizations have made halfhearted attempts at building "knowledge bases," in a hope that some of this institutional knowledge can be committed to electronic paper, preserved, and made more accessible. The problem is that IT professionals aren't necessarily good writers, so the act of producing the knowledge base is difficult for them. It also takes time—time the organization is often unwilling to commit, especially in the face of other daily pressures and demands.

As I said, this is a problem that's difficult to fix. The IT team realizes it's a problem, and is generally willing to fix it—but they're not tech writers, and often have a limited ability to fix the problem. You can usually create management requirements that require problems and solutions be logged in a Help desk ticketing system, but searching through that system for problems and solutions can often be difficult and time-consuming—much like searching for solutions on an Internet search engine, with all of the false "hits" such a search generally produces.

But we *must* find a way to address this problem. Knowledge about the company's infrastructure—and how to solve problems—*has* to be captured and preserved. This requirement is crucial not only to solving problems faster in the future but also to eventually preventing those problems by making better IT management decisions.

How Truly Unified Management Can Fix the Problems

This book is going to be all about fixing these four problems, and the means by which I'll propose to do so falls under the umbrella term *unified management.* Essentially, unified management is all about bringing everything together in one place.

We'll break down the silos between IT disciplines, putting everyone onto the same console, getting everyone working from the same data set, and getting everyone working *together* on problems. We'll do that in a way that brings users, IT, and management into a single viewport of IT service and performance. We'll create more transparency about things like service levels, letting users *see* what's happening in the environment so that they're more informed.

We'll inform users in a way that's meaningful to them rather than using invisible, back-end technical metrics. We'll rebuild the entire concept of SLAs into something that's meaningful *first* to users and management, and that can withstand the transition to "hybrid IT" that's being brought about by outsourcing certain IT services to "the cloud."

Finally, we'll find a way to capture information about our environment, including solutions to problems, to enable faster time-to-resolution when problems occur. In addition, this information will enable management to make smarter decisions about future technology directions and investments.

We'll try to do all of this in a way that won't cost the organization an arm and a leg nor take half a lifetime to actually implement. That will involve a certain amount of creativity, including looking at outsourced solutions. The idea of an outsourced solution providing monitoring for in-sourced components is fairly innovative, and we'll see what applicability it has.

I should point out that much of what we'll be looking at can work to support the IT management frameworks that many organizations are adopting these days, including the ITIL framework that's become popular in the past few years. You certainly don't have to be an ITIL expert to take advantage of the new processes and techniques I'll suggest—nor do you even have to think about implementing ITIL (or any other framework) if your organization isn't already doing so. If you are using a framework, however, you'll be pleased to know that everything I have to propose should fit right into it.

Summary

This chapter has established the four main themes that will drive the remaining chapters in this book. These core things represent what many experts believe are the biggest and most fundamental problems with how IT is managed today, and represent the things that we'll focus on fixing throughout the remainder of this book. Our focus will be on changing management philosophies and practices, not on simply picking out new tools—although new tools may be something you'll acquire to help support these new practices.

Chapter 2 will focus on the first problematic practice, which is the fact that IT tends to be managed in domain-specific silos. We'll look at the technical reasons organizations have been more or less forced to manage this way, and explore ways in which you can start to change that practice.

Chapter 3 will look at connecting people: IT management, your users, your service desk, and more. Only by bringing *everyone* into the process can IT better align itself to the needs of the organization.

Our third problem practice will be the subject of Chapter 4, where we dive into looking *outside* the data center for monitoring. The goal will be to solve the problems we've discussed in this chapter, further focusing IT on its value to the organization.

Chapter 5 will discuss ways to turn problems into future solutions. Although modern organizations are fully aware of the need for Help desk tracking and knowledge building, *how* those activities are managed as part of the larger IT management process can make a huge difference in their value-add to the organization.

We'll conclude in Chapter 6, with an attempt to visualize an IT environment where these new, unified management practices are in place. I'll provide narratives from several case studies, helping you see how these modernized practices work in a real environment.

Download Additional Books from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this book to be informative, we encourage you to download more of our industry-leading technology books and video guides at Realtime Nexus. Please visit http://nexus.realtimepublishers.com.

Realtime
publishers