Realtime publishers

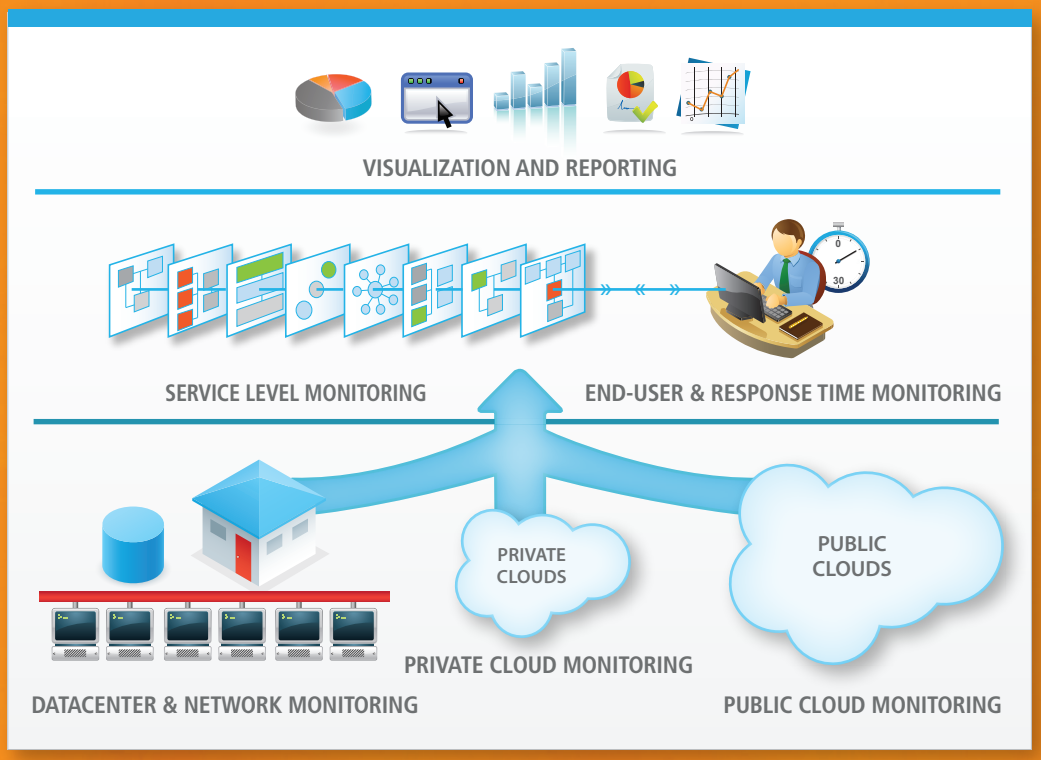
The Definitive Guide™ To

Monitoring the Data Center, Virtual Environments, and the Cloud

sponsored by

Don Jones

The Nimsoft Monitoring Solution



- **Ensures business service delivery regardless of IT platform**
- **Enables rapid adoption of new computer infrastructure such as private and public cloud**
- **Monitors the datacenter to the cloud, including SaaS, hosted, and virtualized environments**
- **Lowers TCO by 80% and delivers proven value in weeks**

Unified Monitoring

nimsoft
From the Datacenter to the Cloud

www.nimsoft.com

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

Chapter 1: Evolving IT—Data Centers, Virtual Environments, and the Cloud

In the beginning, data centers were giant buildings housing a single, vacuum tube-driven computer, tended to by people in white lab coats whose main job was changing the tubes as they burned out. Today's data centers are so much more complicated that it's like a completely different industry: We not only have dozens or hundreds or even thousands of servers to worry about, but now we're starting to outsource specific services—like email, spam filtering, or customer relationship management (CRM)—to Web-based companies selling "Software as a Service (SaaS)" in "the cloud." How do we manage it all, to ensure that all of our IT assets are delivering the performance and service that our businesses need?

Evolving IT

Every decade or so, the IT industry pokes its toes into the waters of a new way of computing. I'm not talking specifically about the revolving thin client/thick client computing model that comes and goes every few years; I'm talking about major paradigm shifts that take place because of radical new technologies and concepts. Shifts that permanently change the way we do business. In some cases, these shifts can resemble past IT techniques and concepts, although there are always crucial differences as we move forward. This is how IT evolves from one state to another, and it's often difficult and complex for the human beings in IT to keep up.

Remember When IT Was "Easy?"

I started in IT almost two decades ago—that's several lifetimes in technology years. When I started, we had relatively simple lives—my first IT department didn't even have a local area network (LAN). Instead, our standalone computers connected directly to an AS/400 located in the data center, and that was really our *only* server. IT was incredibly easy back then: Everything took place on the mainframe. We didn't worry about imaging our client computers because we ultimately didn't care very much about them. Security was simple because all our resources were located on one big machine, and the only connections to it were basically video screen and keyboard feeds. Monitoring performance was incredibly straightforward: We called up an AS/400 screen—I think the command was WRKJOB, for "work with jobs"—and looked at every single IT process we had in a single place. We could bump the priority on important jobs or depress the priority on a long-running job that was consuming too many cycles.

Ah, nostalgia.

Distributed Computing: Flexible, But Tough to Manage

We soon made the move into distributed computing. Soon, we had dozens of Novell NetWare servers and Windows NT servers in our expanding data center. Our computers were connected by blazing-fast Token Ring networks. We shifted mail off our AS/400 onto an Exchange Server. For the first time, our IT processes were starting to live on more and more independent machines, and monitoring them—well, we didn't actually monitor them. If things were a bit slow, there wasn't much we could do about it. I mean, the network was 16Mbps and the processors were Pentiums. "Slow" was kind of expected. And, at the time, the best performance tool we had was Windows' own Performance Monitor, which wasn't exactly a high-level tool for managing anything like Service Level Agreements (SLAs). Our basic SLA was, "If it breaks, yell a lot and we'll get right on it. We have a pager."

That's the same basic computing model that we all use today: Bunches of servers in the data center, connected by networks—100Mbps or better Ethernet, thankfully, rather than Token Ring—and client computers that we have to spend a significant amount of time managing. Gone are the days of applications that ran entirely on the mainframe; now we have multi-tier applications that run on our clients, on mid-tier servers, and in back-end databases. Even our "thin client" Web apps are often multi-tier, with Web servers, application servers, and database servers participating.

We're also more sophisticated about management. Companies today can use tools that monitor each and every aspect of a service. For example, some tools can be taught to recognize the various components—middle-tier, back-end, and so forth—that comprise a given application. As Figure 1.1 shows, they can monitor each aspect of the application, and let us know when one or more elements are impacting delivery of that application's services to our end users.

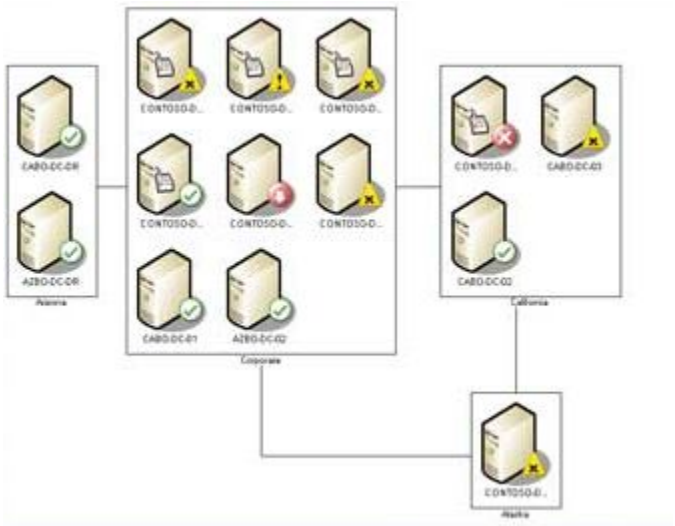


Figure 1.1: Monitoring elements in an application or service.

We've mastered distributed computing, and we have the means to monitor and manage the distributed elements quite effectively. To be sure, not every company *employs* these methods, but they're certainly available. So what's next?

Super-Distributed Computing: Massively Flexible, Impossible to Manage?

The common theme behind all of today's distributed elements is that they live in *our* data centers. Location, however, isn't as important as what our own data centers provide us—absolute control. For every server in our data center, we're free to install management agents, monitor network traffic, and even stick thermal sensors into our servers if we want to. They're *our* machines, and we can do anything with them that, from a corporate perspective, we want to.

But we're starting to move outside of our own data centers. What marketing folks like to broadly call "the cloud" is offering a variety of services that live in someone *else's* data center. For example—and to define a few terms—we can now choose from:

- **Hosted services**, such as hosted Exchange Server or hosted SharePoint Server. In most cases, these are the same technologies we *could* host in our own data center, but we've chosen to let someone else invest in the infrastructure and to bear the headache of things like patching, maintenance, and backups.
- **Software as a Service,** or SaaS, such as the popular SalesForce.com or Google Apps. Here, we're paying for access to software, typically Web-based, that runs in someone else's data center. We have no clue how many servers are sitting behind the application and we don't care—we're just paying for access to the application and the services it provides. Typically, these are applications that aren't available for hosting within our own data center, even if we wanted to, although they *compete* with on-premises solutions that provide the same kind of services.
- **Cloud computing,** which, from a strict viewpoint, doesn't include either of the previous two models. Cloud computing is a real computing platform where we install *our own applications*, often with our own data in a back-end database, to be run on someone else's computers. Cloud computing is designed to offer an "elastic" computing environment, where more computing resources can be engaged to run our application based on our demand. Cloud apps are more easily distributed geographically, too, making them more readily-available to users all over the world.

All of these services are provided to us by a company of some kind, which we might variously call a *hosting provider* or even a *managed service provider* (MSP). Ultimately, this is still the same distributed computing model we've known and loved for a decade or more. We're just moving some elements out of our own direct control and into someone else's, and often using the Internet as an extension to our own private networks. This new model is increasingly being referred to as *hybrid IT*, meaning a hybridization of traditional distributed computing, in conjunction with this new, super-distributed model that includes outsourced services as a core part of our IT portfolio.

But there's the key phrase: *Out of our own direct control.* Without control over the servers running these outsourced services, how can we manage them? We can't exactly install our own management agents on someone else's computers, can we? And for that matter, do we really *need* to monitor performance of these outsourced services? After all, isn't that what the providers' SLAs are for—ensuring that we get the performance we need? These are all questions we need to consider very carefully—and that's exactly what we'll be doing in this chapter and throughout the rest of this book.

Three Perspectives in IT

The world of hybrid IT consists of three major viewpoints: the IT end user, or the person who is the ultimate consumer of whatever technology services your company has; the IT department, tasked with implementing and maintaining those services on behalf of the end user; and the IT service provider, which is the external company that provides some of your IT services to you. It's important to understand the goals and priorities of each of these viewpoints, because as you move more toward a hybrid IT model, you'll find that some of those priorities tend to shift around and change their importance.

The IT End User

The IT end user, ultimately, cares about getting their job done. They're the ones on the phone telling *their* customers, "sorry, the computer is really slow today." They're the ones who don't ultimately care about the technology very much, except as a means of accomplishing their jobs.

Here's something important:

> The IT end user has the most important perspective in the entire world of business technology because without the end user's need to accomplish their job, nobody in IT has a job.

I'm going to make that statement a manifesto for this book. In fact, I'll introduce you to an IT end user (whose name and company name have been changed for this book) that I've interviewed. You'll meet him a few times throughout this book, and I'll follow his progress as his IT department shifts him over to using hybridized IT services.

> Ernesto is an inside sales manager for World Coffee, a gourmet coffee wholesaler. Ernesto's job is to keep coffee products flowing to the various independent outlets who resell his company's products. Like most users, Ernesto consumes some basic IT services, including file storage, email, and so on. He also interacts with a CRM application that his company owns, and he uses an in-house order management application. Ernesto works on a team of more than 600 salespeople that are distributed across the globe: His company sells products to resellers in 46 countries and has sales offices in 12 of those countries.

> Ernesto's biggest concerns are the speed of the CRM and order management application. He literally spends three-quarters of his day using these applications, and much of his time today is spent waiting on them to process his input and serve up the next data-entry screen. His own admittedly informal measurements suggest that about one-third of that time—just under 2 hours a day—is spent waiting on the computer. He knows exactly how much he generates in sales every hour, and since he's paid mainly on commission, he knows that those 9 hours a week—almost a quarter of his work time—are costing him dearly.
>
> He complains and complains to his IT department, as does everyone else, but feels that it's mostly falling on deaf ears. The IT guys don't seem to be able to make things go any faster. There's talk now of outsourcing some of the applications Ernesto uses, such as the CRM application. Ernesto just hopes it doesn't run any *slower*—he can't afford it.

If you work in IT, you know how common a scenario that is. Nothing's *ever* fast enough for our users, and it can be incredibly difficult to nail down the exact cause of performance problems—so we tend to file them all in the "like to help you, but can't, really" folder and go on with our other projects. It'll be interesting to see this same situation from the perspective of Ernesto's IT department.

The IT Department

The IT department, on paper, cares about supporting their users. You and I both know, however, that what IT *really* cares about is technology. Tell us that "email is slow" and we're less interested because that's a big, broad topic. We need to narrow it down to "the network is experiencing high packet loss" or "the email server's processor is at 90% utilization all the time" before we can start to solve the problem. We think in those technical terms because we're *paid* to; interfacing with end users—who typically can't provide anything more definitive than "it seems slower than yesterday" can be challenging.

And so we create SLAs. Typically, however, those SLAs are *not* performance-based but rather are *availability*-based. "We promise to provide 99% uptime for the messaging application, and to respond within 2 hours and correct the problem within 8 hours when the application goes down." That means we can have up to 87.2 hours of downtime—two full work weeks—and still meet our SLA! 99% sounded good, though, and hopefully nobody will think to do the math. But we still haven't addressed *slow* messaging performance because it's difficult to measure. What do we measure? How long it takes to send a message? How long it takes to open a message? What's good performance—5 seconds to open a message? Honestly, if you've ever had to actually wait that long, you were already drumming your fingers on the mouse. A second? That seems like a tough goal to hit. And how do you even measure that? Go to a user's computer, click "Open," and start counting, "one one-thousand, two one-thousand, three one-thou… oh, there, it's open. That's about two and a half seconds."

Instead we tend to measure performance in terms of technical things that we can accurately touch and measure: Network consumption, processor utilization, memory utilization, internal message queue lengths, and so on. Nothing the end user cares about, and nothing we can really *map* to an end-user expectation—how does a longer message queue or higher processor consumption impact the time it takes to open a message?—but they're things we can *see* and take action on, if necessary.

> John works for World Coffee's IT department, and is in charge of several important applications that the company relies upon—including the CRM application and the in-house order management application.
>
> John has set up extensive monitoring to help manage the IT department's SLAs for these applications. They've been able to maintain 99.97% availability for both applications, a fact John is justifiably proud of. The monitoring includes several front-end application servers, some middle-tier servers, and a couple of large back-end databases—one of which replicates data to two other database servers in other cities. John primarily monitors key metrics for each server, such as processor and memory utilization, and he monitors response times for database transactions. He also has to monitor replication latency between the three database servers. Generally speaking, all of those performance numbers look good. As an end-point metric, he also monitors network utilization between the front-end servers and the client applications on the network. He doesn't panic until that utilization starts to hit 80% or so, which it rarely does. When it does, he's automatically alerted by the monitoring solution, so he feels like he has a pretty good handle on performance.
>
> The company's users complain about performance, of course, but the client application has always run fine on John's own client computer, so he figures the users are just being users.
>
> The company plans to start moving the CRM application to an outsourced vendor, probably using a SaaS solution. They also plan to move the in-house order management application into a cloud computing platform, which should make it easier to access from around the world, and help ensure that there are always computing resources available to the application as the company grows. John is relieved because it'll mean all this performance management stuff will be out of his hands. He just needs to make sure they get a good SLA from the hosting providers, and he can sit back and relax at last.

The IT Service Provider

As we start to move to a world of hybridized IT, it's also important to consider the perspective of the IT service provider—the person responsible for whatever IT services are being hosted "in the cloud." These folks have a unique perspective: In one way, they're like an IT department, because they have to manage a data center, monitor performance, patch servers, and do everything else a standard IT department would do. Their "customers," however, aren't internal users employed by the same company. They don't get to use the term *customers* in the same touchy-feely, but ultimately meaningless, way that standard IT departments do. An IT service provider's customers are *real customers,* who pay real money for services—and when someone pays money for something, they expect a level of service to be met. So service providers' SLAs are much more serious, legally-binding contracts that often come with real, financial penalties if they're not met.

Service providers are also in the unusual position of having to expose some of their IT infrastructure to their customers. In a normal IT department, the end users—or "customers," if you like—don't usually care about technology metrics. End users don't care about processor utilization, and might not even know what a "good" utilization figure is. With a service provider, however, the customer *is* an IT department, and they know *exactly* what some of those technology metrics mean—and they may want to know what they are from moment to moment. At the very least, a service provider's customers want to see metrics that correspond to the provider's SLA, such as metrics related to uptime, bandwidth used, and so forth.

> Li works for New Earth Services, a cloud computing provider. Li is in charge of their network infrastructure and computing platform, and is working with World Coffee, who plans to shift their existing Web services-based order management application into New Earth's cloud computing platform.
>
> Li knows that he'll have to provide statistics to World Coffee's IT department regarding New Earth's platform availability, because that availability is guaranteed in the SLA between the two companies. However, Li is worried because he knows most of World Coffee's end users already think their order management application is slow. He knows that, once the application is in the cloud, those "slow" complaints will start coming across *his* desk. He needs to be able to prove that his infrastructure and platform are performing well so that World Coffee can't pin the blame for slowness on him. He knows, too, that he needs to be able to provide that proof in some regular, automated way so that World Coffee has something they can look at on their own to see that the New Earth platform is running efficiently. He knows his customers aren't asking for that kind of detail yet—but he knows they will be, and he doesn't yet know how he's going to provide it.

IT Concerns and Expectations

With those three perspectives in mind, let's look at some of the specific concerns and expectations that each of those three audiences tend to have. This is a way of summarizing and formalizing the most important points from each perspective so that we can start to think of ways to meet each specific expectation and to address each specific concern. Think of these as our "checklists" for a more evolved, hybrid IT computing model.

IT End Users

As I stated previously, IT end users ultimately care about getting their jobs done. That means:

- They expect their applications to respond more or less immediately. They may *accept* slower responses, but the *expectation* is that everything they need comes up pretty much instantly.
- They expect applications to be available and stable pretty much all the time. This is often referred to as *dial tone availability,* because one of the most-reliable consumer services of the past century was the dial tone from your land telephone—which typically worked even if your home's power was out.

And you know what? That's about it. Users don't tend to have complex expectations—they just want everything to be immediate, all the time. That may not always be *reasonable,* but it's certainly straightforward.

IT departments, as a rule, have never done much to manage this expectation, which is why many end users have a poor perception of their IT department. IT has, in fact, found it to be very difficult to even formally define any alternate expectations that they could present to their users.

IT Departments

IT departments tend to have availability as their first concern. Performance is important, but it's often somewhat secondary to just making certain a particular service is up and running at all. In fact, one of the *main* reasons we monitor performance at all is because certain performance trends allow us to catch a service *before it goes down*—not necessarily before it becomes unacceptably slow, but before it becomes completely unavailable. We also tend to monitor technology *directly,* meaning we're looking at processor utilization, network utilization, and so on. So you can summarize the IT department's concerns and expectations as follows:

- They want to be able to manage technology-level metrics, such as resource utilization, across servers.
- They want to be able to map raw performance data to thresholds that indicate the health of a particular service—such as knowing that "75% processor utilization" on a messaging server really means "still working, but approaching a bad situation."

- They want to be able to track performance data and develop trends that help predict growth.
- They want to be able to track and manage uptime and other metrics so that they can comply with, and report on their compliance with, internal SLAs.
- They typically want to be able to track all the low-level metrics associated with a service. For example, messaging may depend on a server, the underlying network, and infrastructure services such as a directory, name resolution, and so on, as well as infrastructure components such as routers, switches, and firewalls.

IT departments, in other words, are end-to-end data fiends. A good IT department—in today's world, at least—wants to be able to track detailed performance numbers on each and every element of their data center, right out to the network cards in client computers, although they typically stop short of trying to track any kind of performance *on* client computers. The theory is that if everything inside the data center is running acceptably, then any lack of performance at the client computer is the client computer's fault.

IT Service Providers

IT service providers have, as I've stated already, a kind of hybrid perspective. They need to have the same concerns as any IT department, but they have additional concerns because *their* customers—other IT departments—are technically savvy and spending real money for the services being provided. So in addition to the concerns of an IT department, a service provider has these concerns and expectations:

- They need to be able to provide performance and health information about *their* infrastructure *to their customers.*
- In many cases, slow performance at the customer end may be due to elements on the customer's network, which is out of the service provider's control. Service providers need to be able to quantify performance of their infrastructure so that they can defend themselves against performance accusations from their customers.
- They need to be able to prove, in a legally-defensible fashion, their compliance with the SLAs between themselves and their customers.
- They need to be able to communicate certain health and performance information to their customers so that customers have some visibility into what they're paying for.

It's actually kind of unfair to service providers, in a way. Most IT departments would *never* be expected to provide, to their own end users, the kind of metrics that the IT department expects from their own service providers.

Business Drivers for the Hybrid, Super-Distributed IT Environment

Let's shift gears for a moment. So far, we've talked mainly about the perspectives and expectations of various IT-centric audiences. As we move into a hybrid IT environment, with some services hosted in our own data center and others outsourced to various providers, meeting those expectations can become increasingly complex and difficult.

But what does the *business* get out of it? IT concerns aside, why are businesses driving us toward a hybrid IT model? I can assure you that if the business didn't have some vested interest in it, we wouldn't be doing it; outsourcing services is never completely free, so the business has to have some kind of ulterior motive. What is it?

Increased Flexibility

Flexibility is one of the big drivers. Let me offer you a story from my own experience from around 2000, when the Internet was certainly big but nothing called "cloud computing" was really in anyone's mind.

> Craftopia.com (now a part of Home Shopping Network) was a small arts and crafts e-tailer based in the suburbs of Philadelphia, PA. The company's brand-new infrastructure consisted of two Web servers and a database server (which, in a pinch, could be a third Web server for the company's site), hosted in an America Online-owned data center (with tons of available bandwidth). The company generally saw fewer than 1,000 simultaneous hits on the Web site, and their small server farm was more than up to that task.
>
> One day, the IT department—all four people, including the CTO—was informed that the company was up for a feature segment on the Oprah television show. Everyone gulped because they knew Oprah could generate the kinds of hits that would melt their little server farm, even though that level of traffic would likely only last for a few days or even hours. If the servers could manage to stay up, they might pull in a lot of extra sales, but not enough to justify adding the dozen or so servers needed to meet the demand. Especially since that surge in demand would be so short. The servers didn't manage to stay up: It was a constant battle to restart them after they'd crash, and it made for a long few days.
>
> Had all this taken place in 2010, the company could simply have put its Web site onto a cloud computing platform. The purpose of those platforms is to offer near-infinite, on-demand expansion, with no up-front infrastructure investment. You simply pay as you go. Sure, the Oprah Surge would have cost more—but it would presumably have resulted in a compensating increase in sales, too. Once the Surge was over, the company would simply be paying less for their site hosting, since the site would be consuming fewer resources again. There wouldn't be any "extra servers" sitting around idle, because from the company's perspective, there weren't any servers at all. It was all just a big cloud.

That's the exact argument for cloud computing, as well as for SaaS and even hosted services: Expand as much as you need without having to invest any infrastructure. If you've grown *just beyond* one server, you don't have to buy another whole server—which will sit around mostly idle—just to add a tiny bit of extra capacity. The hosting provider takes care of it, adding just what you need.

Faster Time-to-Market

Today's businesses need to move faster and faster and faster, all the time. It used to be that taking a year or more to bring a new product or service to market was fast enough; today, product life cycles move in weeks and months. If you need to spin up a new CRM application in order to provide better customer service, you need it *now,* not in 8 months.

With hosted services and SaaS, you can have new services and capabilities in *minutes.* After all, the provider has already created the application and supporting infrastructure; you just need to pay and start using it. This additional flexibility—the ability to add new services to your company's toolset with practically zero capital investment and zero notice—is proving invaluable to many companies. They no longer have to figure out how their already-overburdened IT department will find the time to deploy a new solution; they simply provide a purchase order and "turn on" the new solution as easy as flipping a light switch.

Pay As You Go

Massive capital investment is something that companies have long associated with IT projects. Roll out a major solution like a new Enterprise Resource Planning (ERP) or CRM application, and you're looking at new servers, new network components, new software licenses, and more. It's an expensive proposition, and in many cases, you're investing in capacity that you won't be using immediately. In fact, a quick survey of some of my industry contacts suggests that most data centers use about 40 to 50% of their total server capacity. That means companies are paying fully double what they need simply because we all know you have to leave a little extra room for growth. You want Exchange Server, and you have 500 users, but think you'll have 1500 within 3 years? Well, then you spend for 1500.

That's why the "pay as you go" model offered by service providers is so attractive. If you need 500 mailboxes today, you pay for 500. When you need 501, you pay for 501. It's possible that what you eventually pay for all 1500 will cost more than if you were hosting the service in your own data center, but the point is that you didn't have to pay for all 1500 all along. If you were wrong about your growth, and only needed 1000 mailboxes, then you're not paying for the excess one-third capacity. Pay as you go means you don't have to plan as much, or as accurately, and you're less likely to pay a surcharge for overestimating. Pay as you go lets you get started quickly, with less up-front investment.

Business Goals and Challenges for the Hybrid IT Environment

If there are business-level drivers for hybrid IT, there are certainly business-level challenges to go with them. Remember, we're talking about the *business* here, rather than specific IT concerns. These are the things that a business executive will be concerned with.

Centralizing Management Information

One major concern is where management information will come from. Today, *many* businesses are already getting IT management information from too many separate places and tools. Managers are often forced to look at one set of reports for Microsoft portions of the environment, for example, and a separate set for the Unix- or Linux-based portions.

When some services move out of the data center and into "the cloud," the problem becomes even more complex. In some cases, there's a concern about whether management information will even be *available* for the outsourced services; at the very least, there's an expectation that the outsourced services will be yet another set of reports.

What kind of management reports are we talking about? Availability, at one level, which is a high-level metric but is still important to know. Managers need to know that they're getting what they paid for, and that includes the availability of in-house services as well as outsourced ones.

At another level, consumption is important. Some companies may need to allocate service costs—whether internal or external—across business units or divisions. In other cases, managers need to see consumption levels in order to plan for growth and the accompanying expenses. A definite business goal is get all this information *in one place,* regardless of whether a particular service is hosted in-house or on a provider's network.

Redefining "Service Level"

Businesses really need to redefine their top-level SLAs. Rather than worrying so much about uptime—which seems to be the primary focus of most of today's SLAs—businesses should manage to the *end-user experience* (EUE). In other words, regardless of the service's basic availability, how is it performing from the perspective of the end user? If end users are spending half their time waiting for a computer to respond, the company is potentially wasting a lot of money on that service, regardless of where it's hosted.

This sounds complicated, but that's only because I—and probably you, since you're an IT person—tend to start thinking about the underlying technology. "Do we start guaranteeing a transaction processing time in the database? Do we guarantee a certain network bandwidth availability?" Nope. We guarantee a specific EUE. For example, "When you search for a customer by name, you will receive a first page of results within 3 seconds." You need to identify key tasks or transactions *as seen from the end user perspective,* and write an SLA that sets a goal for a specific time to complete that task or transaction *from the end user's perspective.*

If you're not able to meet that SLA, *then* you dive into technology metrics like network bandwidth, processor utilization, and database response times; the end metric that you drive to is what the end user actually experiences on their desktop. That may sound impossible to even measure, let alone guarantee, but as you move into a hybrid IT environment, it's absolutely essential—and most of the rest of this book will talk about how you'll actually achieve it.

Gaining Insight

IT departments—and thus, the business—have the *option* to get as much insight as they need into their existing data centers. That is, plenty of tools and techniques exist, although not every business chooses to utilize them. Going forward, businesses are going to *have* to have deep insight into the technology assets, because that insight is going to be the only way to achieve that EUE-based SLA that businesses need to establish.

Hybrid IT makes this vastly more complicated to achieve. If your EUE isn't where you want it to be with a cloud-based application, where do you start looking for the problem? Is it the Internet connection between your Taiwan office and the Paris-based cloud provider data center? Is it processing time within your cloud-based application? Is it response time between the cloud-based application server and the cloud-based back-end database? Or is it network latency within the Taiwan office itself? The term *hybrid IT* is an apt one because you're never truly outsourcing the *entire* set of elements that comprise an IT service: Some elements will always remain under your control, while other elements—like the public Internet—may be out of the control of both you and your service provider. You're going to need tools that can give you insight into every aspect so that you can spot the problem and either solve it or adjust your EUE expectations accordingly.

Maintaining Responsibility

Here's another major business challenge in hybrid IT: *It's still your business.* Let's consider a brief section from Amazon's EC2 SLA (you can read the entire thing at http://aws.amazon.com/ec2-sla/):

> AWS will use commercially reasonable efforts to make Amazon EC2 available with an Annual Uptime Percentage (defined below) of at least 99.95% during the Service Year. In the event Amazon EC2 does not meet the Annual Uptime Percentage commitment, you will be eligible to receive a Service Credit as described below.

They define a year as 365 days of 24-hour days, meaning the service can be unavailable for up to about 5 hours a year. However, *if they don't meet that SLA,* you're only entitled to a service credit:

> If the Annual Uptime Percentage for a customer drops below 99.95% for the Service Year, that customer is eligible to receive a Service Credit equal to 10% of their bill (excluding one-time payments made for Reserved Instances) for the Eligible Credit Period. To file a claim, a customer does not have to have wait 365 days from the day they started using the service or 365 days from their last successful claim. A customer can file a claim any time their Annual Uptime Percentage over the trailing 365 days drops below 99.95%.

That means you can't even file a claim until you've had more than 5 hours of outage in a 365-day period. If you do file a claim, you're eligible to receive a *credit*—not a refund—of up to 10% of your bill. I'm not trying to pick on Amazon.com, either, because most service providers in this part of the industry have very similar SLAs.

My point, rather, is that the SLA is not protecting *your business.* If you have a mission-critical application hosted by a service provider, and that application goes down, *your business is impacted.* You're losing money—potentially tens of thousands of dollars an hour, depending on what service is impacted. The SLA is never going to pay for that damage; at best, it's going to refund or credit a portion of your provider fees, and that's all.

The moral is that *you need to remain responsible for your entire business, and all the services you rely upon to operate that business.* You may outsource a service, but you can't outsource responsibility for it. You need to have insight into its performance levels and availability, and you need to be able to engage your service provider *when things start to look bad,* not when they go completely awful or offline. This can be a major challenge with some of today's service providers—and most of them know it, and are struggling to provide better metrics to those customers who demand it. You need to be one of those customers who demands it, because it's your business that's on the line.

Special Challenges for IT Service Providers

If you're an IT service provider with a hosted service, SaaS offering, or even a cloud computing platform, then you know the difficult situation that you're in. On the one hand, you're an IT department. You have data centers, and you need to manage the performance and availability of those resources that are under your control. When things slow down or problems arise, you need to be able to quickly troubleshoot the problem by driving directly toward the root-cause element, whether that's a server, a network infrastructure device, a network connection, a software problem, or whatever.

On the other hand, you're providing a service to a technically-savvy customer—typically, another IT department that's paying for the services you provide. Unlike end users, *your* customer is accustomed to highly-technical metrics, and they're used to having complete control and insight over IT services because those have traditionally been hosted in the customer's own data center. Just because they're moving service elements out of their data center doesn't mean they want to give up all the control they're accustomed to. In fact, smart customers will demand deep metrics so that they can continue to manage an EUE-based SLA. They'll want to know when slowdowns are on their end, or when they can hold you responsible and ask you to work on the problem.

Competitively, you *want* to be the kind of provider that can offer these kinds of metrics and this kind of insight and visibility. As the world of hybrid IT grows more prevalent and more accepted, it will also grow more competitive—and providers that can become a seamless extension of the customer's IT department and data center will be the preferred providers who earn the most money and trust from their customers.

So start thinking: How can you provide your customers with deep metrics in a way that only exposes the metrics *related to them* and not information related to *other* customers? How can you provide this information in a way that will integrate with your customers' existing monitoring tools so that they can treat your services as a true extension of their own data center rather than as yet another set of graphs and reports that they have to look at?

The Perfect Picture of Hybrid IT Management

Let's talk about what the perfect hybrid IT world might look like. This is the pie-in-the-sky view; for now, let's not worry about what's possible but rather focus on what would be best for the various IT audiences and for the business as a whole. We'll use this "perfect picture" to drive the discussion throughout this book, looking at whether this picture is achievable, and if so, how we could do so. If there are any instances where we realize that this perfect picture isn't yet fully realized, we can at least outline the capabilities and techniques that need to exist in order to make this dream a reality.

For IT End Users

Remember, for end users, getting the job done is the key. And while they sort of naturally expect everything to be instant and always-available, we *can* reset that expectation if we explicitly do so in terms they can understand and relate to.

> Ernesto has been using the company's newly-outsourced applications for several months now, and he's quite satisfied with them. The company has published target response times for key tasks, such as locating a customer record in the CRM application and processing a new order in the order-management application.
>
> On Ernesto's computer—and the computers of several of his colleagues across the world—is a small piece of software agent that continually measures the response times of these applications as Ernesto is using them. The information collected by that agent is, he's told, forwarded to his company's IT department, which compiles the data and publishes the actual response times from across the company as an average. Anytime Ernesto feels that the application is slow, he can visit an intranet Web page and see the actual, measured performance—often times, he realizes, the "slowdown" is more his impatience at getting a big new order into the system. On a couple of occasions, though, he's noticed the measured response times falling below the published standard, and he's called the help desk. They've always known about the problem before he called, and were able to let him know which bit of the application was slow, and about when he could expect it to return to normal. He hasn't the slightest idea what any of that means, but it feels good to know that the IT department seems to have a handle on things.

There are actually numerous ways to measure the end user experience—and better ways don't require any kind of agent to be installed on actual end-user computers. That's something we'll explore in later chapters of this book.

For IT Departments

The IT department serves as a human link between the end users and the technologies those users consume. Rather than holding themselves accountable to standards that only they can interpret and understand, however, they're now setting goals that the end users—their "customers"—can comprehend. Fortunately, they're also able to manage to those goals, even across services that are outsourced. By having the right tools in place, the IT department can treat outsourced services just like any other element of the IT portfolio.

> John was concerned about setting SLAs based on end user experience, but because they started with real-world measurements, and used those as the performance baseline, he's found that he no longer has to fend off as many "things are slow" complaints. End users know what kind of performance to expect, and so long as John provides that performance, they're satisfied if not always delighted.
>
> He was especially worried about providing those SLAs for services that were outsourced. However, World Coffee now receives a stream of performance metrics directly from their hosting providers. When things are slow at the end user computer, John can see exactly where the slowdown is occurring. Sometimes it's in the communication between networks, and John can bug his ISP about their latency. Sometimes it's the communication within the provider network, between database and application server, and John can call their help desk and find out what's going on. They're defining new, performance-based SLAs with the providers, which will help ensure that the provider is engineering *their* network to keep up with demand.

For IT Service Providers

Service providers *want* to do a good job for their customers—after all, that's what earns new business, retains business, and grows business relationships. They're discovering that the way to do that is not always by being a "black box" but by offering some visibility. After all, customers are betting a portion of their business on the provider, and they deserve a little insight into what they're betting on.

> Li's work with World Coffee is going well. Thanks to the detailed metrics he's able to provide them, and because they're using a single tool to monitor their entire service portfolio, they tend to call only when there's legitimately a problem on Li's end. Best of all, the same tools that provide customers like World Coffee with data are also providing *him* with performance information, helping him spot declining performance trends *before* actual performance starts to near the thresholds that might trigger an SLA violation.

About this Book

This chapter has really been an introduction, with a goal of helping you to understand the goals and challenges you face as you evolve your IT services to a hybrid IT model. We've outlined the evolution from today's IT models to the future's super-distributed, hybrid model, and covered some of the key concerns and problems you're likely to face as you move along that path. What we need to cover—and what the rest of this book is about—is how you actually accomplish it.

Chapter 2 will dive into the issue of monitoring in some detail. I want to look at how companies monitor their IT environments today, and discuss how they probably *should* be monitoring those same environments—because we all know that not every environment is doing all they can in terms of service monitoring! But then I'll look specifically at why today's accepted practices really start to fall apart when you move into a hybrid IT model, and explore new goals that we can set for monitoring as our IT environment moves toward that super-distributed model.

In Chapter 3, I'll propose a new model for defining SLAs internally. This isn't a radical new model by any stretch, but in the past, it's been impractical to achieve. I want to really lay out what we should be looking for in terms of IT service levels, and look at some of the techniques that we can employ to do so—right now, not years in the future.

Chapter 4 is an acknowledgement that although the EUE is a great top-level metric, it doesn't actually help us solve performance problems. We still need to be able to dive into performance at a very detailed, very granular component level—but how can you accomplish that in a world where half of your "components" live on someone else's network and are even abstracted away from the hardware they run on? I'll propose some capabilities for new monitoring tools that can help not only solve the super-distributed challenge but also streamline your everyday troubleshooting processes and procedures.

Chapter 5 is the real-world, nitty-gritty look at what you're going to need to successfully manage a hybrid IT environment, where you've got services hosted in your own data center as well as in someone else's. Although I'm not going to compare and contrast specific vendor tools, I will provide you with a shopping list of capabilities so that you can start engaging vendors and truly evaluating products with an eye toward the value they bring to your environment.

Chapter 6 is going to take this idea of hybrid IT monitoring and move it up a level in the organization, proposing the idea that IT health reporting can become just another one of the services you offer to your customers—such as managers. I'll also spend some time covering this topic from the perspective of an IT service provider company, when their customers truly are paying customers, and where management reporting becomes a significant value-add.

We've got a lot of ground to cover, but I think this is one of the most important topics that IT faces as we begin hybridizing our IT environments. Sure, issues like security and user access are important, but in the end, we need to be able to ensure that these outsourced IT services can support our businesses. That's what this book is all about.

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.