InfoWorld

GET TECHNOLOGY RIGHT®

IT Strategy Guide

State of the Enterprise Architecture

INSIDE

Compliments of: **f5**

Introduction

EACH OF THE FOLLOWING SEVEN ARTICLES TACKLES A key characteristic of emerging enterprise architectures, as perceived by *InfoWorld* contributors and editors.

But in reality, we probably could have boiled them down to one: the business-driven architecture.

No matter which design buzzword you choose these days — SOA (service-oriented architecture), ILM (information lifecycle management), virtualization, and so on — chances are there's a business driver behind it. Never before has the business side had such a direct, quick, and forceful impact on how IT environments are being designed and refreshed. And never before have CIOs — and even CEOs — been so interested in architecture.

"Architectural principles have been raised to the level of the CIO, and I'm the keeper of the architecture," asserts John Halamka, CIO of both Harvard Medical School and CareGroup Health System, a hospital group with 3,000 physicians. Halamka runs a monthly architecture meeting for all CareGroup developers and waxes eloquent about new health-care regulations that make sound architecture a business imperative.

"We're charged with knowing where our information lives, who's touched it, and why it was touched," Halamka says. "How can you do this without a data-centric, services-centric, and highly reliable architecture?"

Gartner Group Vice President Jeff Schulman agrees. "Over the last few years," he says, "85 percent of the work of architects has been in the physical and logical layers, but we're now moving to 50 percent in the business layer." Schulman sees this shift to business-driven thinking as a sign of what the future holds for IT: "We need people who can understand and relate key business issues to their likely impact on IT architecture."

For a taste of how this is translating into architecture, read on. And if you disagree with any one of our categories, just substitute in the words "business-driven." You'll be fine. ☙

Data-Centric Architectures

ENTERPRISES HAVE ALWAYS BEEN CONCERNED with data quality and integration. But the interest in improving data and content management is clearly on the rise, as companies are increasingly focusing on unifying their enterprisewide data and on designing architectures to maximize the usefulness and accessibility of that data.

The reasons are at least twofold. First, the costs of error-ridden, inconsistent, and obsolete data are high, in terms of slowing business processes and hindering automation. Second, business leaders are keen to take more information into account — either structured or unstructured, from both transactional and content systems — when making decisions, and too much information remains locked away in silos.

For many large companies, a data-centric architecture starts with rationalizing the "master data" — the identities and attributes of customers, products, employees, and other core reference data — at the heart of the business. In a global enterprise, customer or product data is typically spread across dozens, even hundreds, of implementations of CRM, ERP, and other systems, often from different vendors.

Each set of data is typically tailored to a specific business need — engineering, sales, or marketing — and location. The result, from the top-down view, is a sea of fragmented data that leads inevitably to faulty BI.

The emerging class of master data management solutions from Oracle, SAP, Siebel, and other enterprise application vendors attempts to bring order to this chaos. Oracle's Enterprise Data Hubs, for example, combine a publish-and-subscribe mechanism, process automation based on configurable rules, and a knowledge base that helps data managers reconcile differences among source systems. Some solutions, such as Siebel's, throw in business analytics capabilities. But all master data management solutions aim to create a canonical master data set that gets pushed to all kinds of data repositories — mainframes, transactional systems, data warehouses — throughout the organization.
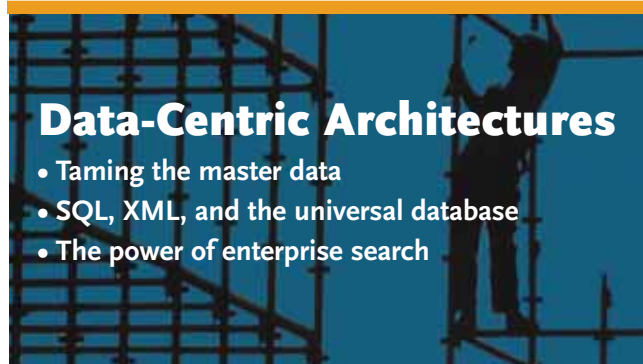
The goal is not merely to synchronize data across systems but to improve data quality and to deliver as a service accurate, consistent data to transactional and operational systems. "It isn't simply a matter of connecting the plumbing between many different data sources," says Robert Shimp, vice president of technology marketing at Oracle. "There's a quality function that has to be applied, to clean, dedupe, and reconcile all of this information. You don't just need data; you need services-based information."

In addition to mastering the master data, enterprises are also beginning to bridge the gaps between structured and unstructured data sources, as new technologies and techniques — especially XML, SOAs, and enterprise search — are making it easier and less expensive to do so. IBM's WebSphere Information Integrator, for example, can combine SQL-, object-, and content-oriented access methods — as well as enterprise search techniques — to perform queries across relational databases, XML stores, mainframes, file servers, content management systems, even e-mail systems.

According to Eric Sall, IBM Software Group's program director of information integration, the benefits go beyond

the obvious operational advantages, such as a user of a CRM application being able to view an open trouble ticket in the customer service system. The pervasive, on-the-fly querying capabilities of enterprise search also extend the capabilities of traditional BI to include real-time data not yet loaded into the data warehouse.

Looking ahead, Oracle's Shimp thinks this universal approach to searching and reporting will eventually put

Data-Centric Architectures
- Taming the master data
- SQL, XML, and the universal database
- The power of enterprise search

the data warehouse to pasture. The key enablers here are databases that can store relational data and native XML together.

"Traditionally, people have had to load and unload data, cleanse it and reformat it, do all kinds of complex gyrations, add all kinds of banks of servers for separate OLAP or data mining applications," Shimp says. "That's all going away. We're simplifying down to just a core database that can handle all of this directly inside the database engine."

It will take some time before we reap the full benefits of services-based information and universal data access. OASIS and other standards groups continue their work to establish the core identities and semantics within vertical industries and across them so that companies can more effectively share information through XML. Meanwhile, the walls between database silos, application silos, and organizational silos are coming down.

As IBM's Sall puts it, "You can't be compliant in a silo. You have to be able to look across silos to have any prayer of being compliant as an organization. Same thing with business intelligence. You don't want to be intelligent about a silo, and not about the silo next to it. These are the reasons why this kind of more holistic or enterprise view of information is beginning to be such a big issue with the industry."

— *Doug Dineley*

Process-Driven Architectures

TAKE A LOOK AT SOME OF THE ACRONYMS CREEPING INTO enterprise architectures — BPEL (Business Process Execution Language), BPM (business process management), BPO (business process outsourcing) — and you begin to see a pattern. The "business process" concept has entered the collective IT consciousness in a big way, thanks to a strong focus in the corporate world on efficiency, speed to market, and compliance.

"CIOs are making sure their people are becoming more process-centric and are looking to make decisions on the application architecture that will let them be more flexible with their business processes," says Trevor Naidoo, managing director at German software company IDS Scheer.

Forrester Senior Vice President Merv Adrian says today's business process focus is part of a long-term evolution in which IT thinks about design and architecture at an increasingly higher level. "With each generation, we've raised the level of abstraction, and today the level we're pointing at is the visual business process," he says.

Adrian sees IT increasingly developing, optimizing, and managing systems from the perspective of the end-to-end business process, rather than from silo-oriented transactional efficiency or other resource-driven standpoints. "How quickly do we get the cars off the boat and into the dealer lots?" he asks, adding that this kind of thinking will "focus everybody in the IT organization on what their contribution to business value is."

What exactly is a business process? Some typical end-to-end processes include: order-to-cash, procure-to-pay, product development, and HR-related procedures. "And within those there are a lot of subprocesses," Naidoo notes. Another important set of processes are IT operational processes themselves, as described by the ITIL (IT Infrastructure Library) framework.

Hewlett-Packard Software CTO Russ Daniels explains that businesses tend to focus their IT investments on improving the processes that give their business the most leverage — for example, cutting product design time from three years to 18 months in the auto industry.

What does it take to design IT architectures from a business process perspective? IT must start with process strategy, then design, then execution, IDS Scheer's Naidoo says. The business side must first formally define its processes, HP's Daniels agrees, for process-driven architectures to succeed.

"The degree to which you can automate IT capabilities depends on the degree of clarity the customer has around their processes," Daniels says. "If the business isn't willing to put in the work and make trade-offs, it's very difficult."

Next, IT must be able to model and implement a process-driven architecture at an abstracted level, enabling rapid change as processes change and thereby providing more flexibility to the business. "We have to break the binds that glue IT to a particular set of resources," says EMC CTO Jeff Nick, "so you can flexibly reconfigure, redeploy, and introduce new resources almost as objected-oriented components that are composed together to represent the business process being served."

To this end, software companies ranging from small specialist shops such as IDS Scheer to large players such

as SAP are building BPM tools that overlay existing environments to help model, optimize, and monitor the performance of key business processes from both a resource and workflow perspective.

Often these tools rely on Web services or similar interfaces to enable end-to-end connectivity across an entire environment in support of a business process. And most

Process-Driven Architectures

- The shift away from an operation-oriented paradigm
- The rise of graphical process modeling tools
- Web services and end-to-end process development

support emerging Web services standards such as OASIS' BPEL, which helps describe business processes and their interconnections.

Increasingly, IT architectures must also incorporate knowledge about process best practices, IDS Scheer's Naidoo adds. "We've made a commitment to provide more content in the tools," he says, citing prebuilt reference and process models such as the Supply Chain Council Reference Model, which includes standard key performance indicators. "It's almost like a process factory. ... Pull together and assemble best practices into your process architecture."

— David L. Margulius

Secure Architectures

THANKS TO COMPLEX PERIMETERS, SOPHISTICATED application-level threats, and regulations that hold CEOs and CIOs accountable for company data, security must now be regarded as more than a bunch of technologies tacked onto the network. "Companies are realizing they must approach security at the enterprise level," says Rich Caralli, senior member of the technical staff at the CERT Coordination Center's survivable enterprise management group. "Rather than chasing the latest threat, they're working on identifying and securing directly the core business processes and information assets essential to the company mission."

In fact, security is moving so deeply into business processes and infrastructure that it may someday disappear as a category unto itself. "As organizations develop more mature capabilities in business and IT processes, they're seeing the significant security benefits," Caralli says. "They're moving beyond just patch management, for example, to configuration management or availability management." No matter what you do in the enterprise, he adds, your function is likely to include information security.

If anything has eliminated the effectiveness of traditional perimeter security, it's the growth of anytime, anywhere access.

"Companies used to depend on employees carrying around a notebook with a VPN for remote access," says John Pescatore, vice president of Internet security at Gartner. "But with SSL VPNs, employees now access the network from home computers, other companies' comput-

ers, kiosks, Kinko's, and cybercafés."

Pescatore recounts that at a recent RSA conference, he stepped up to a kiosk that displayed a venture capitalist's e-mail revealing that company X was in for $2.1 million. Or, to take a less spectacular example, a dab of keyboard-logging spyware on a system at FedEx Kinko's can easily capture an employee's password and send it to an attacker. And because telecommuters often share their home computers, a little laxity by Junior as he downloads music on the same machine can breed infection, thereby compromising the corporate network.

Enterprises have also wrestled with the security implications of outsourcing, connections with partners and suppliers, and the growth of Web services. "They can't just hope that the call center in India handling customer data or the company they outsource payroll or sales-force automation to understands and meets their security requirements. They must take into account the impact of an attack or infection on that company's network," Pescatore says. The same goes for connected partners and suppliers.

"No doubt Web services and applications reduce the value of traditional firewalls, as companies have to expose data to the world," says Johannes Ullrich, CTO of The SANS Institute's Internet Storm Center.

With more application-layer exploits sneaking past traditional firewalls as legitimate port 80 traffic, companies have turned to application-savvy intrusion prevention solutions, as well as application layer and XML firewalls placed strategically to protect specific sensitive data. In fact, these capabilities are increasingly merging with tra-

Secure Architectures

- The twilight of point security solutions
- The rise of enterprisewide identity
- New demand for application-layer security

ditional firewall solutions.

"By 2006, when someone buys a firewall, they'll also be buying intrusion prevention," Pescatore says. He also sees XML and Web application security merging with content-filtering products, as evidenced by F5 Networks' purchase of Magnifire.

Web services and compliance requirements are also driving the need for end-to-end enterprise identity solutions and federated identity standards that allow different organizations to set up trust relationships with one another. Identity management's centralized auditing function, in particular, is becoming an important compliance tool.

Traditional desktop and network management solutions have increasingly taken on patch management and other security functions. Their hardware and software inventory capabilities have also become essential components of a viable security strategy, as PC-based technologies and Web servers have been incorporated into a variety of devices.

"My wife worked for a company that sold oscilloscopes running Windows 2000," SANS' Ullrich says. "Did you patch your oscilloscope today?" Switch vendors such as Cisco are working security into their mainstream network hardware. "Each port in your Cisco switch is a perimeter that you can shut down when a security event happens," Ullrich says.

Finally, companies are working security into the development and implementation process much earlier. "Outside code review and vulnerability and penetration testing have become more widespread," Ullrich says. Caralli agrees, "It's much better to head off the security threat much earlier in the process, before you inherit it in the operations phase." The result is that security is on its way to being part of everything else. "In the work we're doing, we're really trying to lose the term 'security,' " Ullrich says.

— *Leon Erlanger*

Lightweight and Open Architectures

YOU'VE HEARD THE SAYING. "NOBODY EVER GOT fired for buying IBM." That mantra was first popularized in the 1970s, long before the advent of the personal computer changed the face of enterprise IT. Big iron was king — often in the form of IBM mainframes — and enterprise software was a big, big deal.

In those days, the Free Software Foundation was barely a glimmer in Richard Stallman's eye. A full suite of enterprise applications meant an expensive, long-term contract, preferably with the largest (and hence most reliable) vendor you could find — somebody like IBM, for example.

A lot has changed since then, but some things remain the same. IBM still wants to be the go-to vendor for every enterprise's IT needs. To secure that position in the 21st century, however, it's had to learn to play a new ball game to suit today's market.

"The size of the marketplace and the kinds of customers that we can address with technology is growing and moving into nontraditional areas," says Doug Heintzman, director of technical strategy at IBM's software group. "There's a whole set of SMBs and new businesses and startups that, quite frankly, haven't traditionally been IBM customers, that have different kinds of thresholds."

What those businesses have in common is a need for greater flexibility and agility than traditional, monolithic IT infrastructures can provide. They're also more risk-averse when it comes to IT expenditures. A smaller shop won't pay for a big, expensive software suite that's full of features it will never use, especially if it anticipates changing market conditions ahead.

What's more, these concerns aren't limited to SMBs and startups. Increasingly, even enterprise customers are demanding low-cost, low-overhead, flexible architectures that offer scalable performance without threatening to hamstring IT agility as a result of vendor lock-in.

At the root of this trend is the proliferation of mature, open, industrywide technology standards. Open standards level the playing field by enabling interoperability between competing products in a given software category, allowing customers to choose freely from among different vendors' offerings. Standards also open the door for the open source community to create its own implementations in key software categories, which drives down customer IT costs even further.

"In many ways, we view open source, in many situations, as 'open standards on steroids,'" Heintzman says. A standard merely describes a common protocol or format, but an open source implementation brings it to life.

IBM isn't the only major software vendor to embrace open source and open standards as a means of appeasing agility-conscious customers; Hewlett-Packard, Novell, Oracle, and many others have joined suit. When Computer Associates made its Ingres relational database open source last year, it very quickly realized a double benefit that lent new luster to a product that had previously been merely a reliable but unremarkable performer.

"It's been very difficult for people to move data off their existing databases into other forms," explains Tony Gaughan, senior vice president of product development at CA. By making Ingres open source, not only did CA give customers a very real cost incentive to switch data-

bases, it also opened the door for the open source community to add features. One of the first major contributions to the product was an engine that allowed Ingres to understand Oracle's proprietary PL/SQL query language, making it much easier for Oracle customers to migrate their applications to Ingres.

This trend toward openness, standardization, and flexibility isn't limited to software. As Sun Microsystems can testify, customers are turning away from large, high-powered single machines in favor of scalable clusters of commodity, Intel-powered 1U and blade servers. Google is perhaps the greatest example of this new kind of architecture, boasting a datacenter composed of tens of thousands of PC-based servers coupled with fault-tolerant software.

True, IBM still sells mainframes; but its sales pitch has changed considerably since the 1970s. These days, IBM zSeries boxes run Linux in addition to z/OS, and main-

Lightweight and Open Architectures
- Industry standards and the undermining of vendor lock-in
- Open source across major software categories
- The rise of big-vendor support

frame processor units are billed as a way to quickly deploy virtual servers in a clusterlike configuration, using IBM's z/VM virtualization software.

From the datacenter to the desktop, scalability, flexibility, openness, and standardization have become the watchwords of the new IT. The message for vendors: Watch out. If your products aren't competitive, it may well be you who ends up getting fired.

— *Neil McAllister*

Pervasive Architectures

ASK 20 PEOPLE WHAT A PERVASIVE IT ARCHITECTURE IS and you'll get 20 answers. Some will mention the rapid spread of always-on wireless and mobile devices. Others will focus on the coming proliferation of billions of tiny, IP-connected sensors, RFID tags, and monitoring devices. And some will think in terms of time rather than space, equating the concept of pervasive with around-the-clock operation.

But however you slice it, recent technology advances are pushing businesses into a world where anytime, anywhere access to people, applications, and data is becoming crucial to success — and it's clear that IT must design systems from the ground up with these requirements in mind.

"I have 2 million square feet of wireless coverage," says Dr. John Halamka, CIO of both CareGroup Health System and Harvard Medical School. "I went for 100 percent coverage throughout the hospital so that I could enable workflow — anytime, anywhere access to data for clinicians who are truly mobile knowledge workers."

"It's all about getting information to the point of business when you need it," agrees Danny Shader, CEO of mobile software maker Good Technology. "Everyone's building networks on the assumption that it'll be IP everywhere," he says, noting that technologies such as Wi-Fi, Wi-Max, 3G, and 2.5G virtually guarantee an always-connected business world. "It's not yet reliable, cheap, secure, or free, but its everywhere — people have spent billions on this," Shader adds.

"It's the death of the business day," says Jeff Schulman, vice president of architecture at Gartner. "The old model was shutdown and batch catch-up; now it's seven by 24.

There's a dynamism here that really pushes on our architectures. In real-time mode, there are responsiveness issues and issues around capacity."

Schulman also claims that in this pervasive future, everything will be addressable and the state of everything will be fully known. "If there is this level of profound connectivity, from an architectural standpoint, there's a very different level of managing resources, of understanding state, of process optimization, and even of governance," he says. "And there's lots of privacy and security issues to be wrestled with."

How should enterprise architects prepare for this coming pervasive future? "Pervasive is a side effect of doing everything Web-exposed and middleware-driven," CareGroup's Halamka says. "My programmers these days are experts in the glue, which is what allows us to create what feels like an integrated product even though the parts may be very different."

In other words, leverage a unified back end to serve multiple channels, devices, and formats. "Most CIOs would like to have a pervasive presence for their customers that's consistent from an internal operations point of view, says Mike McCue, CEO of Tellme Networks, a provider of VoiceXML and VoIP-based outsourced services. McCue claims that, with 4 billion users worldwide, the phone is still the most pervasive device in existence and should be served not from proprietary voice systems but from the same IP-based Web infrastructure that can support tracking, personalization, and other rich functionality.

"The mobile device is the tail and not the dog," agrees Good Technology's Shader. "People don't want to deploy

Pervasive Architectures

- The proliferation of handheld wireless devices
- Events-generating RFIDs and sensors
- The end of business hours and the rise of always-on

something custom, they just want to get a better R on the I they've already made — and the tools for doing that are standard operating systems, Web services, and composite applications."

Willy Chiu, vice president of high-performance on-demand solutions at IBM, adds that transformation technologies such as XML content style sheets should be built into architectures in anticipation of global deployments requiring different languages and domain-specific content. "Whether it's for an insurance broker doing claims adjustment, a broker trading on the stock market floor, or a real estate agent, pervasive systems can customize to those devices based on an underlying middleware layer, and transforming and assembling fairly open piece parts," Chiu says.

— *David L. Margulius*

Service-Oriented Architectures

TO UNDERSTAND AND APPLY THE PRINCIPLES OF SOA, you'd think we would have to agree first on what we mean by a "service." To a surprising degree, we haven't, but this is hardly the first time a powerful idea has been tricky to nail down. Definitions of "objects' and "components" — the ideas that powered earlier phases of software's evolution — were just as elusive.

Writing recently for ACM Queue, ObjectWatch CEO Roger Sessions offered one useful way to think about these successive waves of technology. All three models are ways of packaging code for reuse, he suggests. They differ in terms of where and how the code runs. Objects share a common operating system process and execution environment — for example, Linux, Windows, Java, or .Net. Components live in different processes but share an environment. Services cross both process and environment boundaries.

The environment for Web services and SOA is the global Internet. Of course, that's been true for quite a while. A decade ago programmers began using the Web's Common Gateway Interface to publish and consume services. When we build and deploy services today — using REST (Representational State Transfer) and XML-over-HTTP on the one hand, or SOAP, WSDL, and the WS-* specs promoted by Microsoft and IBM on the other — we build on that common heritage. SOA extends the tradition along two axes: data representation and data communication.

Everyone agrees XML is the lingua franca of data representation, but there's lively debate about how to use it. XML Schema, for example, is an optional feature that sharply divides communities of practice. Do inter-operable services require strict formal data definition or do they require fuzziness? The perplexing answer is both — at different times, in different ways, for different purposes.

In the world of SOAP and WS-*, XML Schema typically governs the contracts between services. If the XML document that represents a purchase order isn't a valid instance of the relevant schema, it's time to throw down the warning flag. And with XML Schema, any process, running anywhere — even offline — can perform that validity check. Let's say that while flying to Chicago you use an InfoPath form to create a purchase order and then e-mail it to the approver when you land. The approver can focus on the business aspects of the order, secure in the knowledge that he or she has received and will relay to the order processing service a document that will be acceptable to that service.

What about the stuff that won't fit into the schema? Today this contextual data travels in e-mail, where we can't do much with it. Defining parts of schemas that can carry arbitrary XML content, so people can "scribble in the margins," is a key strategy. At the same time, don't ignore the growing amounts of XML data flowing through your enterprise that is not, and may never be, schematized. The prime example is RSS. All kinds of useful services, done in the REST and XML-over-HTTP style, are coming up from the grassroots. We think of RSS mainly in terms of blogging, but it also affords us a lightweight and incredibly versatile way to exchange, route, and recombine all kinds of stuff. Nearly every application that today uses e-mail to connect people and processes

Service-Oriented Architectures

- **Strategy and packaging for code reuse**
- **XML Schema and RSS communication under one roof**
- **The promise of new Web services protocols**

can be recast as an RSS-oriented service. Easier and more robust integration, no spam — what's not to like?

In fact, this low-tech approach is so appealing that many people are now discounting the WS-* stack. That's understandable and in many cases valid. While we argue about which WS-* standards will stick to the wall, a set of key capabilities is emerging. Broadly speaking, WS-* pushes aspects of data communication — security, asynchrony, reliability, routing, and proxying — up into the application layer where we can reason about these things as businesspeople rather than wrestle with them as network plumbers.

That's a lofty statement, but here's a concrete example to nail it down. Let's say your order processing service is used by a dozen applications and by hundreds of people. Suddenly, one morning, it's triple-witching time: You add a new application, you implement a mandated auditing rule, and then you have to reroute traffic because a server fails. On days like that it won't ever be easy to get home by dinnertime. The set of principles embodied in an SOA, however, may at least make it possible.

Cynics will note that we've been enumerating those principles for a couple of years now. You've heard the litany: coarse-grained messages, loosely coupled processes, data-driven integration, self-describing data, programming-language and platform neutrality, pervasive intermediation. We call this cluster of ideas by different names — grid, enterprise service bus, service-oriented architecture. It's quite possible that next year's favorite acronym won't be SOA. But many if not most of the ideas will survive — and will define the dominant style of enterprise software for years to come.

— *Jon Udell*

Needs-Based Architectures

SAY GOODBYE TO ONE-SIZE-FITS-ALL ARCHITECTURES — THE old days of just giving users their apportioned slice of "the system" are long gone. As financial pressures force IT departments to act more like internal service businesses, architectures are by necessity becoming more responsive to the fast-changing needs and service-level requirements of multiple segments of users.

"It's the idea of IT as a service company and needs-based, service-level management that needs to get built into the architecture," explains Gartner Vice President Jeff Schulman. "We're saying, think about what's really driving the need for architecture in your organization right now. What are the business needs?"

The übertrend in application design, SOA, has made serving disparate sets of needs easier in some senses because application functionality can be accessed using standard protocols, rather than having to be built from scratch. On the other hand, it's very difficult to spec out an SOA with the current and future needs of all stakeholder in mind.

CareGroup and Harvard Medical School CIO John Halamka echoes this advice, citing numerous examples where his company's IT architecture was designed to support varying service levels based on internal customer needs.

"We were one of the earliest adopters of information lifecycle management," Halamka says, noting that the company's 85-plus terabytes of data are stored on a tiered storage architecture based on the data object's point in the lifecycle and value to end-users.

Halamka also points out that CareGroup's wireless infrastructure was designed to support four very different classes of users. Physicians get the highest speed and most secure service for clinical results and ordering. Patients use the same network but with a different security model. And the network supports two other applications: VoIP over wireless and RFID for geolocation of people and assets, each with their own distinct service levels.

"What I've done is looked at the needs and then architected those systems to serve those differing levels of needs," Halamka says. "You've got to take a hierarchical view; you can't deliver the absolute highest level of service and quality across the board because our capital budgets are limited."

Taking this approach to the extreme, some enterprises have designed systems that allow them to fine-tune service levels to highly specific segments of internal users or external customers. Charles Schwab, for example, has developed an architecture that allows it to provide segmented levels of service to its online customers, according to Willy Chiu, vice president of high-performance on-demand solutions at IBM.

"You do get a better page response the more money you have in your account," Chiu says. "Before the market opens, for example, they cache the top 500 users." Such segmentation is typically policy-driven and can be done dynamically. "This is differentiation of service, class of service; what the business side has been talking about all this time," Chiu notes. "But in the old days, the IT guys said, 'We have no way of doing this.' "

For internal users, such segmentation is often role-based and requires an architecture that can support pol-

Needs-Based Architectures

- Enterprise IT as insourced service organization
- Balancing specialized and enterprisewide requirements
- Building extensibility into needs-based systems

icy-setting and rules engines, as well as usage monitoring and charge backs, according to Chiu. On the back end, such systems require flexibility — the ability to expand or contract based on demand or to shift loads dynamically to provide the right service levels to the right users as seamlessly as possible.

Where will this all lead? Gartner's Schulman predicts the ascendancy of what he calls "good-enough" architectures — architectures that are constantly changing to meet the shifting needs of internal and external customers, in contrast to the old days of monolithic moonlaunch style architectural planning.

"We're saying don't go there. ... Architectures must be built to change, not to last," Schulman advises. "Understand what's driving it, build out enough that you can understand it and then modify it. Get something on the board, build out an ROI, and then come back."

— *David L. Margulius*