Your Next IT Strategy

by John Hagel III and John Seely Brown

Harvard Business Review

Reprint R0109G

Harvard Business Review

October 2001

HBR Case Study R0109A
Off with His Head?
David Champion

HBR at Large R0109B
The Leadership Lessons of Mount Everest
Michael Useem

Different Voice R0109C
Genius at Work:
A Conversation with Mark Morris

Harnessing the Science of Persuasion R0109D
Robert B. Cialdini

Torment Your Customers
(They'll Love It) R0109E
Stephen Brown

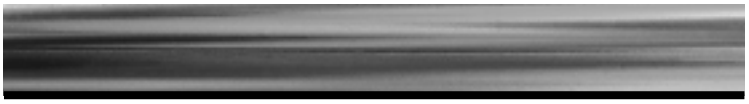
Radical Change, the Quiet Way R0109F
Debra E. Meyerson

Your Next IT Strategy R0109G
John Hagel III and John Seely Brown

HBR Interview R0109H
Bernard Arnault of LVMH:
The Perfect Paradox of Star Brands
Suzy Wetlaufer

Best Practice R0109J
Speeding Up Team Learning
Amy Edmondson, Richard Bohmer, and Gary Pisano

Tool Kit R0109K
Boost Your Marketing ROI
with Experimental Design
Eric Almquist and Gordon Wyner



The age of proprietary information systems is coming to an end, and the age of shared services is dawning. Don't panic. There's a pragmatic way to make the change pay off for your company.

Your Next IT Strategy

by John Hagel III and John Seely Brown

OVER THE PAST YEAR, AS THE HYPE OVER E-COMMERCE HAS subsided, a new chorus of promises about the potential of the Internet has been gaining volume. The singers this time are not dot-coms and their backers but rather the big providers of computer hardware, software, and services. What they're promoting, through a flurry of advertisements, white papers, and sales pitches, is a whole new approach to corporate information systems. The approach goes by many different names – Microsoft calls it ".Net," Oracle refers to "network services," IBM touts "Web services," Sun talks about an "open network environment" – but at its core is the assumption that companies will in the future buy their information technologies as services provided over the Internet rather than owning and maintaining all their own hardware and software.

No doubt, many executives are skeptical. They've heard outsized promises and indecipherable buzzwords before, and they've wasted a lot of time and money on Internet initiatives that went nowhere. This time, though, there's an important difference. The technology providers are not making empty promises: They're backing up their words with massive investments to help create the infrastructure

needed to make the new IT approach work. As these efforts continue, over the next year or two, a steady stream of new, Internet-based services will come on-line, providing significant cost savings over traditional, internal systems and offering new opportunities for collaboration among companies. Slowly but surely, all your old assumptions about IT management will be overturned.

In this article, we will provide an executive's guide to the new IT strategy. We will explain what the Web services architecture is, how it differs from traditional IT architecture, and why it will create substantial benefits for companies. We will also lay out a measured, practical plan for adopting the new architecture – a step-by-step approach that will pay for itself while mitigating the potential for organizational disruption. Indeed, we believe that two of the great advantages of the Web services architecture are its openness and its modularity. Companies won't need to take high-risk, big-bang approaches to its implementation. They can focus initially on opportunities that will deliver immediate efficiency gains, incorporating new capabilities as the infrastructure becomes more robust and stable.

The New Architecture

Until now, companies have viewed their information systems as proprietary. They bought or leased their own hardware, wrote or licensed their own applications, and hired big staffs to keep everything up and running. This approach has worked, but it has not worked well. After years of piecemeal technology purchases, companies have inevitably ended up with a mishmash of disparate systems spread throughout different units. Over the last decade, in efforts to merge these "data silos," many big companies have invested large amounts of money – hundreds of millions of dollars, in some cases – in massively complex enterprise-resource-planning systems, which offer suites of interlinked applications that draw on unified databases. The ERP systems have certainly solved some problems, but they've been no panacea: Most big companies still struggle with a hodgepodge of hundreds of incompatible systems. And ERP systems have also created new problems. Because they're relatively inflexible, they tend to lock companies into rigid business processes. It becomes hard, if not impossible, to adapt quickly to changes in the marketplace, and strategic restructurings, through acquisitions, divestitures, and partnerships, become fiendishly difficult to pull off. In effect, the companies that have installed ERP systems have replaced their fragmented unit silos with more integrated but nonetheless restrictive enterprise silos.

The Web services architecture is completely different. Constructed on the Internet, it is an open rather than a proprietary architecture. Instead of building and maintaining unique internal systems, companies can rent the functionality they need – whether it's data storage, processing power, or specific applications – from outside service providers. Without getting too technical, the Web services architecture can be thought of as comprising three layers of technology, as described in the sidebar "An Overview of Web Services." At the foundation are software standards and communication protocols, such as XML and SOAP, that allow information to be exchanged easily among different applications. These tools provide the common languages for Web services, enabling applications to connect freely to other applications and to read electronic messages from them. The standards dramatically simplify and streamline information management – you no longer have to write customized code whenever communication with a new application is needed.

The service grid, the middle layer of the architecture, builds upon the protocols and standards. Analogous to an electrical power grid, the service grid provides a set of shared utilities – from security to third-party auditing to billing and payment – that makes it possible to carry out mission-critical business functions and transactions over the Internet. In addition, the service grid encompasses a set of utilities, also usually supplied and managed by third parties, that facilitates the transport of messages (such as routing and filtering), the identification of available services (such as directories and brokers), and the assurance of reliability and consistency (such as monitoring and conflict resolution). In short, the service grid plays two key roles: helping Web services users and providers find and connect with one another, and creating trusted environments essential for carrying out mission-critical business activities. The role of the service grid cannot be overemphasized: A robust service grid is vital to accelerating and broadening the potential impact of Web services. Without it, Web services will remain relatively marginal to the enterprise.

The top layer of the architecture comprises a diverse array of application services, from credit card processing to production scheduling, that automate particular business functions. It is this top layer that, day to day, will be most visible to you, your employees, your customers, and your partners. Some application services will be proprietary to a particular company or group of companies, while others will be shared among all companies. In some cases, companies may develop their own application services and then choose to sell them on a subscription basis to other enterprises, creating new and potentially lucrative sources of revenue.

John Hagel III *is the chief strategy officer of 12 Entrepreneuring, an operating company in San Francisco. He can be reached at jhagel@12.com.* ***John Seely Brown*** *is the chief innovation officer of 12 Entrepreneuring, where he developed the perspectives in this article, and also serves as the chief scientist of Xerox. He can be reached at jsb@12.com.*

An Overview of Web Services

The Web services architecture has three layers. The most fundamental layer consists of software standards (such as XML) and communication protocols (such as SOAP and its likely extensions) that make it possible for diverse applications and organizations to do business together electronically.

The middle layer is the service grid, through which specialized utilities provide key services and tools. Four types of utilities operate over the service grid.

Shared utilities provide services that support not only the application services residing in the top layer but also the other utilities within the service grid. For example, security utilities provide such services as authentication, authorization, and accounting. Performance auditing and assessment utilities assure users of Web services that they will obtain agreed-upon levels of performance and will be compensated for damages if performance falls below these levels. Billing and payment utilities aggregate charges for the use of Web services and ensure prompt and full payment.

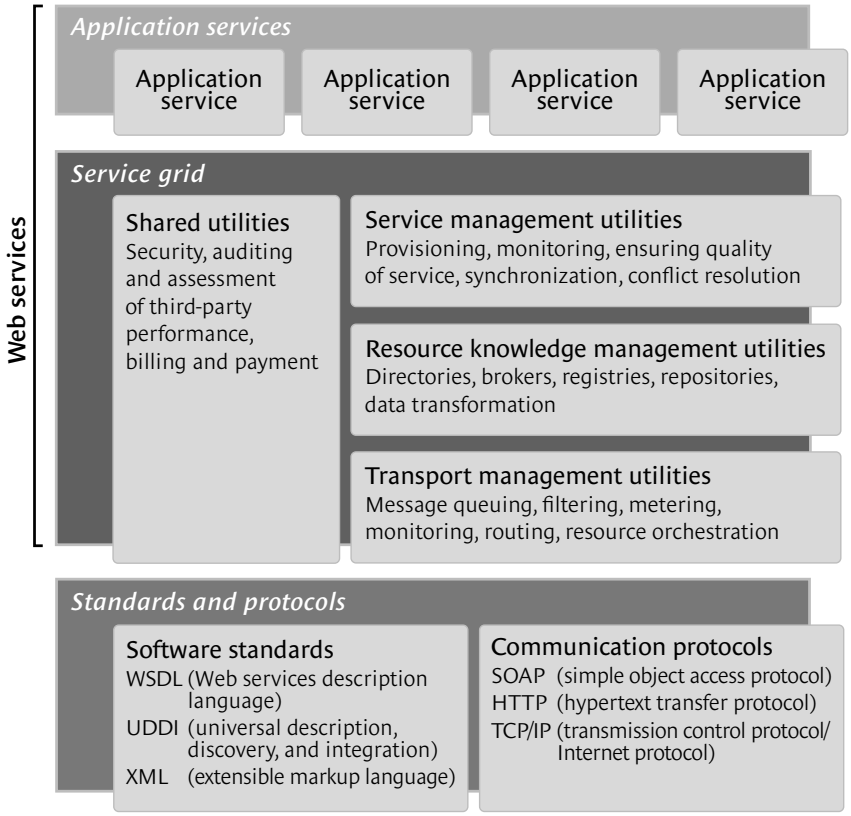
Transport management utilities include messaging services to facilitate reliable and flexible communication among application services as well as orchestration utilities that help companies assemble sets of application services from different providers.

Resource knowledge management utilities include service directories, brokers, and common registries and repositories that describe available application services and determine correct ways of interacting with them. They also include specialized services for converting data from one format to another.

Service management utilities ensure reliable provisioning of Web services. They also manage sessions and monitor performance to ascertain conformance to service-quality specifications and service-level agreements.

The top layer encompasses a diverse array of application services that support day-to-day business activities and processes – everything from procurement and supply chain management to marketing communications and sales.

The Web Services Architecture



To illustrate how the architecture works, let's contrast the way a typical business activity – loan processing by a bank – would be carried out through a traditional proprietary architecture and the Web services architecture. Loan processing is a complex procedure requiring at least six steps (data gathering about an applicant, validation of data, credit scoring, risk analysis and pricing, underwriting, and closing) and involving interactions with a number of other institutions (checking an applicant's credit rating, verifying investment and loan balances, and so on). With a traditional IT architecture, the process is usually supported by one, very complicated application maintained by an individual bank; like a Swiss Army knife, the integrated application does a lot of things, but it may not do any of them particularly well. And since the costs of maintaining electronic connections with other institutions are high, requiring leased communication lines and expensive software to link different systems, the necessary interactions are often handled manually through phone calls and faxes. The process, in sum, is cumbersome, costly, and prone to errors.

With the Web services architecture, loan processing becomes much more flexible, automated, and efficient. Leased lines are replaced with the Internet, and open standards and protocols take the place of proprietary technologies. As a result, the bank can connect automatically with the most appropriate institution for each transaction, speeding up the entire process and reducing the need for manual work. And rather than maintain its own integrated loan-processing system, the bank can take a modular approach, using specialized Web services supplied by an array of providers. It can also shift easily among providers, using one service, say, for risk analysis of loans to restaurants and another for risk analysis of loans to hospitals. In other words, the bank will always be able to use the best tool for the job at hand; it will no longer have to compromise on performance to avoid the complexity of integrating proprietary applications.

Clearly, the Web services architecture offers important advantages over its predecessor. First, it represents a much more efficient way to manage information technology. By allowing companies to purchase only the functionality they need when they need it, the new architecture can substantially reduce investments in IT assets. And by shifting responsibility for maintaining systems to outside providers, it reduces the need for hiring numerous IT specialists, which itself has become a significant challenge for many companies. Using Web services also reduces the risk that companies will end up using obsolete technologies; third-party utilities and application providers will be required to offer the most up-to-date technologies in order to compete. Companies will no longer find themselves stuck with outdated or mediocre applications and hardware. The standardized, plug-and-play nature of such an architecture will also make it much easier for companies to outsource activities and processes whenever it makes economic sense. (See the sidebar "Big Changes for Your IT Department.")

Second, and perhaps more important, the Web services architecture supports more flexible collaboration, both

Big Changes for Your IT Department

The shift to the Web services architecture for corporate computing is not only a matter of adopting new technology. It will require broad organizational and managerial changes as well as the development of new kinds of capabilities. A particularly big impact will be felt in the corporate IT department. CIOs will face new challenges and assume new roles:

- IT departments will need to move in two dimensions simultaneously: outsourcing many traditional IT activities (as credible and reliable providers of services emerge) while leveraging internal capabilities to design distinctive Web services that can be sold to other companies. *CIOs will become strategists and entrepreneurs,* assessing areas of competitive advantage and focusing resources on building new IT-based businesses.
- IT departments will need to integrate new sets of skills in such areas as enterprise application architecture, enterprise application integration, application development, security, and IT operations. *CIOs will become knowledge brokers,* pulling together expertise from within and outside their companies.
- IT operations and performance will increasingly depend on the effective integration of external resources, requiring deeper skills in structuring and managing relationships. *CIOs will become relationship managers,* coordinating the efforts of an array of companies.
- IT departments will need to take the lead in shaping the standards required for industries and business communities to operate effectively. *CIOs will become negotiators,* their leadership styles shifting from command-and-control to persuade-and-influence.

among a company's own units and between a company and its business partners. When traditional information systems need to talk to each other, they do so through dedicated, point-to-point connections. For example, when a sales-force-management application needs to send information on closed sales to a payroll processing application for the computation of commissions, a programmer has to write a special piece of code – a connector – to allow the two systems to communicate. The problem with such point-to-point connections is that they are fixed and inflexible and, as they proliferate, become nightmares to manage. With the Web services architecture, tight couplings will be replaced with loose couplings. Because everyone will share the same standards for data description and connection protocols, applications will be able to talk freely with other applications, without costly reprogramming. This will make it much easier for companies to shift operations and partnerships in response to market or competitive stimuli. The loose-coupling approach of Web services also makes it an attractive option within an organization. CIOs can use the Web services architecture to more flexibly integrate the extraordinarily diverse set of applications and databases residing within most enterprises while at the same time making these resources available to business partners.

Until now, what's been called e-business has for the most part been a primitive patchwork of old technologies. Most companies that do business on the Internet have had to yoke together existing systems with new ones to create the illusion of integration. A visitor using a corporate Web site may think it's a single, streamlined system, but behind the scenes, people are often manually taking information from one application and entering it into another. Such swivel chair networks, as they have come to be known, are inefficient, slow, and mistake ridden. Merrill Lynch, like almost all large companies, has struggled to patch together hundreds of different applications to support its sites for customers. John McKinley, the company's CTO, draws an analogy to the Potemkin villages in czarist Russia, where brightly painted facades hid the unseemly reality of run-down homes. The Web services architecture promises to solve this problem. Taking the people out of the network, the architecture will enable connections between applications – both within and across enterprises – to be managed automatically.

First Steps to Success

The construction of the Web services architecture is still in its early stages, and years of investment and refinement will be required before a mature, stable architecture is in place. This does not mean, however, that companies should wait to begin the transition to a new IT strategy; even today, benefits can be gained by moving to a Web services model for certain activities and processes. But it does mean that companies should take a pragmatic, measured approach. Fortunately, the Web services architecture is ideally suited to such an approach: Because it's based on open standards and it leverages the capabilities of third parties, companies don't have to place big bets at the outset. They can carefully stage their investments, learning important lessons along the way. (See the sidebar "Five Questions You Need to Ask.")

Merrill Lynch's McKinley, for example, is currently leading a number of initiatives designed to take advantage of Web services. One initiative is the creation of an innovative portfolio-analysis system for use by brokers and selected customers. By using XML to link disparate systems

Five Questions You Need to Ask

Senior managers need to ensure that their organizations' executives are thinking ahead about the implications of the Web services architecture. Here are five questions you can use to spur your people:

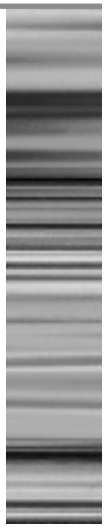
Does our management team have a shared vision of the long-term (five to ten years out) business implications of the new IT architecture?

Do we have a transition plan that balances the state of the architecture's development with a clear understanding of the areas of highest business impact?

Are we moving fast enough today to build our expertise and exploit immediate opportunities for streamlining intercompany processes, outsourcing activities in which we don't have distinctive capabilities and designing Web services that we can market to other companies?

Do we have a clear understanding of the obstacles within our organization that may hinder us from exploiting the full value of the new IT architecture, and do we have initiatives under way to overcome these obstacles?

Are we exerting sufficient leadership in shaping both the functionality offered by providers of Web services (defining, for example, the performance levels required for mission-critical applications) and the standards needed to collaborate with our partners?



within Merrill Lynch as well as to integrate information from partner organizations, the new system will tie together customer information, product information, and real-time market data in a flexible, low-cost way. It will give the company's brokers up-to-the-second, integrated views of all the information they need to meet a customer's needs at any given moment. Merrill Lynch is also using a Web services approach to enable brokers and clients to access information and applications from a wide variety of devices, including computers, PDAs, cell phones, and conventional phones. Both of these projects offer immediate business benefits: They provide an important competitive advantage to the company's salespeople while delivering added value to customers.

Merrill Lynch's experience, as well as that of other early adopters such as General Motors and Dell, offers three guidelines for other companies looking to get a head start.

Build on your existing systems. The Web services architecture should initially be viewed as an adjunct to your current systems. Through a process we call node enablement, you can use Web or application servers to connect your traditional applications, one at a time, to the outside service grid, turning them, in effect, into nodes on the Internet. Node enablement is often as simple as creating an explicit record of the connection specifications of an application – documenting, in other words, its application programming interfaces, or APIs – along with the application's name, its Internet location, and procedures for connecting with it. The existing application is left intact but is "exposed" so that it can be found and accessed by other applications in the Web services architecture. The process of node enablement should be systematic, driven by near-term needs but shaped by a view of longer-term opportunities.

General Motors provides a useful example of this process. Mark Hogan, the president of eGM, a business unit created by the auto giant to oversee its consumer Internet initiatives, is a strong advocate of the Web services architecture. Like Merrill Lynch, eGM began with fairly conventional Web sites connecting the company with customers and dealers. Now, however, Hogan and his team have developed a road map for using Web services to move GM to a dramatically new build-to-order manufacturing and distribution model, which will enable the company to generate added revenue and use its assets much more efficiently. This initiative requires the ability to communicate and collaborate electronically with more than 8,000 dealers, all with information systems of widely differing specifications and sophistication. Few of the dealers have cutting-edge IT skills, and fewer still have the money to invest in major new applications. Given these constraints, says Hogan, "traditional IT architectures simply aren't up to the task. The Web services architecture provides the only way to rapidly enhance our IT platform." By applying node enablement to existing applications at GM and at the dealers, new processes can be rolled out incrementally with relatively modest investment.

For the first stage in the transition, GM is focusing on using Web services to enhance its traditional build-to-stock model, providing a broader set of options for dealers and customers. It has provided dealers, for example, with a locate-to-order functionality – a Web services–based application that quickly finds specific car models in the inventories of other dealers. GM is also planning to roll out an order-to-delivery application, which will shorten the lead time between placing a customer order and delivering the vehicle. Such interim steps will pave the way to offering the ultimate build-to-order model, which will require the reconfiguration of manufacturing operations and a more sophisticated deployment of Web services.

The payoff is expected to be enormous. GM's long-term goal is to cut in half its $25 billion investment in inventory and working capital. Analysts at Goldman Sachs estimate that supply chain initiatives using Web services could ultimately reduce GM's operating cost per vehicle by more than $1,000. Yet the staged approach to change allows GM to shift its IT architecture slowly, avoiding disruption and focusing only on systems that will deliver real economic paybacks at each stage of deployment. It also allows the company to temper the risk involved in moving to a new technology platform, since GM's efforts are tied to the evolution of the architecture.

Start at the edge. In implementing the new architecture, early adopters are concentrating their initial efforts at the edges of their enterprises – on the applications and activities that tie their companies to customers or to other companies. Sales and customer support are obvious examples of edge activities, as are procurement and supply chain management. Less obviously, some traditionally internal functions can be pushed out to the edges as a result of outsourcing. In the electronics industry, for example, many production activities are being contracted to specialized manufacturing service providers, creating a need to share formerly proprietary applications and data.

> GM's long-term goal is to cut in half its $25 billion investment in inventory and working capital.

Why is there so much focus on the edge? Because that's where the limitations of existing IT architectures are most apparent and onerous. Almost by definition, an application on the edge can benefit by being shared. As a result, it suffers most from the difficulties in connecting proprietary, heterogeneous systems. As GM found, rolling out a

new set of applications to its far-flung dealer network was next to impossible before the emergence of Web services.

Dell Computer provides a great example of the benefits of starting at the edge. Dell's relationships with its suppliers of components and other direct materials are critical to the company's strategy. The total amount the company spends on direct materials equals as much as 70% of its revenue, so even modest savings in supply chain costs will have a big impact on the bottom line. A related and equally important concern to Dell is inventory management. In the personal computer industry, where product prices have recently been declining at 0.6% per week, excess inventory can become very costly.

Recognizing the huge gains possible from more effective supply-chain management, Dell focused its early Web services initiatives in this area. It began by more closely connecting its assembly operations with the network of outside logistics providers that operate the distribution centers for direct materials – the vendor-managed hubs, as Dell calls them. Traditionally, the company had to hold substantial inventory in the supply chain to ensure that products could be delivered quickly to customers. Its goal was to fill orders in five days, yet it took suppliers an average of 45 days to fill materials orders. To ensure it did not run short of key components, suppliers had to maintain ten-day inventory buffers at vendor-managed hubs, and Dell had to maintain buffers of 26 to 30 hours at its own assembly plants. In addition, every week, Dell distributed a new 52-week demand forecast to all suppliers.

Today, Dell generates a new manufacturing schedule for each of its plants every two hours, reflecting actual orders received, and publishes these schedules as a Web service via its extranet. Because the schedules are in XML format, they can be fed directly into the disparate inventory-management systems maintained by all the vendor-managed hubs. The hubs always know Dell's precise materials requirements and can deliver the materials to a specific loading dock at a specific building, from which they are fed immediately into an assembly line. With this new approach, Dell has been able to cut the inventory buffers at its plants to just three to five hours. Explains Eric Michlowitz, the company's director of supply chain e-business solutions, "We've been able to remove the stock rooms from the assembly plant, because we now pull in only materials specifically tied to customer orders. This has enabled us to add more production lines, increasing our factory utilization by one-third."

Of course, such lean manufacturing approaches often just push inventory back from the manufacturer to the supplier. Dell's goal, however, is to eliminate excess inventory throughout the supply chain. So the company is now focusing on reducing the buffers held at the hubs. These stocks could be cut substantially if supply problems could be identified earlier. If, for example, Dell knew that one supplier was having a problem fulfilling an order for a particular part, it might be able to temporarily remove from its Web store the computer model that used the part. This would, in turn, enable a reduction in the stocks of the part held at the hubs. To establish such an early warning system for its supply chain, Dell is rolling out an "event management" Web service, again using its extranet. This service automatically sends out queries on the status of orders to suppliers, whose own systems automatically send back responses. Dell expects that this system will reduce hub inventories by as much as 40% while at the same time significantly improving gross margins by better matching demand and supply.

Trying to engage with too many partners too fast is one of the main reasons that so many on-line market makers have foundered: The transactions they had viewed as simple and routine actually involved many subtle distinctions in terminology and meaning.

Create a shared terminology. The move to a shared IT architecture raises an obvious question of control: Who calls the shots? Within a single company, a CIO can impose a set of standards governing information technology (requiring, for example, that accounts always be represented in applications as "ACCTS"). But once a group of companies, each with different internal systems and standards, begins to collaborate electronically, establishing clear lines of authority becomes difficult. In some cases, one company will have the market power to impose standards on its partners, but these situations are rare and, given the increasing complexity and fluidity of corporate partnerships, usually unwise. Instead, shared meaning, and the trust it engenders, must develop much more organically among participants.

Incremental implementation of Web services can aid this process. By starting with a few long-standing business partners – as GM did with its dealers and Dell did with its logistics providers – companies gain room to experiment; they can establish through trial and error a common technical language. Then, as they learn what works and what doesn't, they can expand the orbit of their partnerships to encompass new companies. Trying to engage with too many partners too fast is one of the main reasons that so many on-line market makers have foundered: The transactions they had viewed as simple and routine actually involved many subtle distinctions in terminology and

meaning. That doesn't mean that shared standards can't be established among large groups; it just can't be done easily or overnight. Traditional distributors spend years learning the shades of meaning used by different buyers and sellers. A produce distributor, for example, has to build an understanding of how each of its suppliers quantifies the ripeness of an orange as well as how each buyer evaluates ripeness. It is only then that the distributor will have the knowledge and the authority to create a standard rating system for oranges and promote its adoption throughout the community of buyers and sellers.

XML can be a powerful tool for building shared meaning in Web-based communities, but it's important to understand that XML isn't a cure-all. While XML establishes a common grammar – a framework for sharing meaning – it establishes only very limited semantics. The precise meanings of XML terms still need to be determined by the actual partners. For instance, a particular XML tag may refer to the price of a product, but that doesn't tell you if it's the net price after discounts, if it includes shipping, and so on. Subtleties of meaning have to be hashed out before business can be conducted in all its inevitable complexity. And don't expect the meanings, once established, to stay fixed. They will evolve as partners gain experience and discover shortcomings in their shared processes.

The service grid will play an important role in helping business communities build shared meaning, since a set of utilities will be established to facilitate the development of trading standards. In many cases, the dominant companies within private trading networks will provide these utilities. In other cases, industry consortia will take the lead. RosettaNet is an early example of a consortium-driven utility. It is defining and promoting the adoption of standard XML formats to describe processes in the supply chain of the electronics industry, enabling all participants to use the same terms to describe activities like issuing purchase orders. Such utilities might also be provided by independent businesses that are focused solely on developing XML or other software standards within an industry or across industries.

Shared meaning will naturally increase as the use of the Web services architecture expands. In the architecture's current, early stage of development, incentives for its adoption are limited because relatively few application services are available and the functionality of the service grid is limited. In this period, early movers like Merrill Lynch, GM, and Dell play key roles by providing their business partners with compelling reasons to use Web services. Over time, as additional resources become accessible, the benefits of adopting this architecture will become compelling to more and more companies. Newcomers will find it advantageous to adopt meanings already in use in order to tap into existing applications and utilities.

A Platform for Growth

Although many of the early uses of Web services will focus on reducing costs, efficiency-driven initiatives are only the beginning. Ultimately, the greatest beneficiaries of this new technology will be companies that harness its power for revenue growth. (See the sidebar "Unbundling and Rebundling.")

The new architecture provides, for example, a platform for companies to offer their core competencies as services to other companies. Smart businesses, in other words, won't just consume Web services; they'll also sell them. That's exactly what Citibank is doing right now. It saw that one drawback to early on-line exchanges was the inability to handle payments: Participants would use an exchange to reach agreement on the terms of a transaction but would then have to process payments either manually or through specialized banking networks. Leveraging its deep skill in electronic payments, Citibank quickly introduced CitiConnect, an XML-based payment-processing service that plugs into existing trading applications.

Here's how it works. A company purchasing supplies through an Internet exchange platform, such as one offered by Commerce One, registers information about the authorization levels for specific employees and the corporate bank accounts to be used for payment. When a purchase is made, the buyer clicks the CitiConnect icon on the Web site. An XML message containing payment instructions is automatically assembled, specifying the amount involved, the identity of the buyer, the identity of the supplier, the bank from which to withdraw funds, the bank to transfer the funds to, and the timing of payment. The message is then routed, according to predefined rules, to the appropriate specialized settlement networks.

While XML establishes a common grammar – a framework for sharing meaning – it establishes only very limited semantics. The precise meanings of XML terms still need to be determined by the actual partners.

The benefits for buyers and sellers are compelling: Sellers cut settlement times by 20% to 40%, and both buyers and sellers reduce settlement costs by 50% to 60%.

Unbundling and Rebundling

Two and a half years ago, Marc Singer and I wrote "Unbundling the Corporation" (HBR March–April 1999). In that article, we described how most companies encompass three very different types of businesses – customer relationship management, infrastructure management, and product innovation and commercialization – and how the Internet would facilitate their unbundling, leading to much more tightly focused companies.

The rise of the Web services architecture will not only speed this unbundling but will spur the growth of the new companies by letting them mobilize a greater range of resources to reach a broader set of customers. Freed from the straitjacket of existing enterprise-centric IT architectures, companies won't have to acquire new assets to grow (a slow and often treacherous process); they will be able to rent them, as Web services, from third parties. The capital-intensive model of owning resources will be supplanted by the much more efficient model of orchestrating resources. A new kind of business organization – a rebundler focusing on the assembly and coordination of business processes that stretch across entire industries and markets – will likely emerge and gain enormous power.

Of course, exploiting these new kinds of growth opportunities will require much more than just the new technology architecture. Very different organizational capabilities must be developed – not just new skills but also new performance measurement and reward systems and knowledge management approaches. Even more fundamentally, managers will need to adopt new mind-sets: They will need to focus on creating new opportunities, largely by helping to define and deploy standards, rather than simply trying to adapt to rapidly changing environments.

– John Hagel III

Citibank turned an existing operational capability into a new service line and extended its reach to a broader range of customers through its partnerships with application providers like Commerce One. And Commerce One, for its part, got happier customers while also associating itself with the respected Citibank brand.

The relationship between Citibank and Commerce One illustrates a broader pattern that will both speed the adoption of the Web services architecture and open new growth opportunities for traditional companies. All providers of inter-enterprise applications – private trading exchanges, procurement services, supply-chain management services, and so on – recognize that they need to add new functions rapidly to attract and retain customers. They have two choices: develop the added functionality themselves or source it from specialized providers. Many are choosing the latter path because it's faster and allows them to dedicate their scarce resources to acquiring customers.

In supplying these functions, traditional companies have an important advantage. When an application provider has to choose between sourcing a Web service from a well-known enterprise with a strong track record or from a small start-up with an uncertain future, it will likely go with the established company, as Commerce One did with Citibank. As traditional companies begin to make their capabilities available to other companies through Web services, the importance of the service grid will become increasingly apparent. Web services for functions like invoicing, payments, and logistics are critical to the companies that use them. Without the security, reliability, and performance auditing that service grid utilities can provide to their customers, few enterprises will be willing to offer, much less subscribe to, such mission-critical services.

As the service grid matures and companies move aggressively to exploit revenue growth opportunities created by the Web services architecture, a curious dynamic will begin to play out. The distinction between users and suppliers of Web services will fade. Companies will provide Web services to others in areas in which they have distinctive expertise while at the same time buying Web services from others in areas in which they lack special skills. Over time, the location of particular capabilities – whether inside or outside the walls of any given company – will become less important than the ability to discover and orchestrate distinctive capabilities across enterprises in order to deliver greater value to customers. In the process, many companies will find themselves turned inside out, with their formerly well-guarded core capabilities visible and accessible to all.

The authors thank Dennis Layton-Rodin, Halsey Minor, Mahmoud Falaki, and other colleagues for their contributions to this article.

Reprint R0109G

To place an order, call 1-800-988-0886.