CHAPTER 1

A Day in the Life of a CIO

"The last thing one knows—is what to put first."

Blaise Pascal

"While we ponder when to begin, it becomes too late to do."

Quintillan

It is Monday morning . . . Bob Dunston, a Fortune 500 Chief Executive Officer (CEO), is pondering alternative strategies to spur growth for the next five years. He knows that organic growth from ongoing operations can be improved through a number of programs focused on increasing operational efficiency and productivity. Specifically, it will be critical to reduce cycle times in manufacturing, lower inventory levels, improve supply chain visibility, and enhance customer satisfaction and loyalty. Dunston hopes that he can achieve these goals without the need to invest in another huge enterprise application project. "The ERP project was tough," he says to himself. "I'll retire before I do that again!" Of course, his Information Technology (IT) organization is now consumed with the early phases of an enterprise wide Customer Relationship Management (CRM) project, defining the organization's CRM strategy and performing preliminary vendor evaluations.

The current Research and Development (R&D) pipeline of new products is a bit thin, and until a strategic initiative revamping R&D takes hold, the company will need to pursue a Merger and Acquisition (M&A) strategy. Realizing the M&A challenges that most companies face, he knows that any acquired company must integrate into the parent organization quickly and efficiently to achieve the synergies of the acquisition. With a desire to leave the acquired company's R&D and manufacturing operations intact, Dunston knows that it is critical to initially seek financial integration, followed by

absorption of the remaining business operations such as information technology, human resources, and other centrally leveraged functions. How can he complete this goal quickly and inexpensively while helping to achieve the intent of his M&A strategy? He ponders this, then he recalls a lunchtime conversation with his Chief Information Officer (CIO), who had explained the potential of some rapidly emerging technologies and standards to help improve their supply chain visibility and reduce inventory, as well as enabling other business initiatives, such as M&A integration and procurement processes. He picks up the phone and punches the CIO's extension, muttering to himself, "Let's see if Sedgewick can help with these problems."

Bill Sedgewick scans his calendar for the week, paying particular attention to the pending close of the quarter. He knows that the company's results, although solid, have slowed for the past four quarters. While he is doing the best he can to support the business units with reliable IT solutions, Sedgewick knows that there is untapped potential in the IT organization to drive better business results. When he arrived at the company three years ago, they hadn't had a CIO for two years. In fact, the previous vice president of IT, who worked for the CFO, was an operations guy who formerly ran the data centers. With the rapid pace of IT change and explosive growth of the Internet, his capabilities had been clearly challenged and the business lost faith in the IT function. The CEO hired Sedgewick to fix that situation, and although Bob Dunston was an old-school manager, he was prepared to listen to new ideas.

The phone rings. Sedgewick answers, pressing the speakerphone button. "Hi, Bob. What's up?"

"Bill, what's your day looking like? I'd like to continue that discussion we were having at lunch the other day about—what were they?—Web services; yes, that was it, right? Web services? I've been kicking some ideas around, and I wanted to get your perspective."

He hears Sedgewick shuffling papers and tapping on his keyboard, the staccato clicking of his keys pouring through the phone line like machine gun bursts. "Bob, I'm slammed this morning, but can we catch up later this afternoon, say around four? Does that work?"

"Yes, Bill, that's fine. It's no big deal, but I wanted to finish that discussion in light of some new ideas I've been mulling over. I'll update you at four, okay?" Dunston says.

"Great, see you then, Bob." Sedgewick hangs up the phone and sits back in his chair, wondering to himself, "Hmmm, what's he up to now?"

Later that day, Sedgewick knocks on Dunston's office door. "Hi, Bob. Are you ready?"

"Bill, yes, come on in. How goes the battle today?" Dunston clears his desk as Sedgewick settles into one of the four chairs surrounding the polished table positioned across the office from the CEO's modest, yet contemporary desk.

"Well, you know, we're fighting the good fight. So, what's up?"

Dunston picks up a legal pad, a pen, and his half-full bottled water and eases into a seat across from Sedgewick. "Bill, I want to get back to that Web services discussion we were having the other day. I want to see if there's a fit for some of your ideas on Web services to some of the strategies and business initiatives I think we'll have to embark on over the next two years. Now, you know I've talked about using a few strategic acquisitions to beef up our R&D and product pipeline, right?"

"Yes, Bob, and I've got a team working on an integration strategy that will allow us to very quickly absorb acquisition targets, first for financial reporting purposes, and then complete integration of all order entry, product and customer master data, fulfillment and logistics processes, as well as other key business processes and functions, too."

Sedgewick's face lights up as he explains this point. He is clearly excited about implementing corporate strategy using his information management organization, processes, and capabilities. "We'll have a draft strategy document to show you early next week."

Dunston nods vigorously, clearly eager to hear this information from Sedgewick. "Good, I like that. We're already evaluating M&A targets based on a rigorous profiling methodology, and we should have a short list ready by the end of the month. I want you involved in the due diligence process as we get that short list pulled together."

"In addition, though, you know we're under continued pressure to drive productivity and improve operating margins. I'm concerned with our core businesses—their organic growth is slowing and margins are deteriorating, so I'm looking at a number of programs to reverse these trends. Inventory management could be improved. Reducing cycle time in manufacturing is important. Ultimately, I'd like to shift a large percentage of our business to a make-to-order model, much like Dell's, versus our current inventory-intensive make-to-stock model."

As Sedgewick rapidly scribbles notes, Dunston continues. "Take inventory management, for example. We need better visibility across our entire supply chain to effectively manage inventory. That means tapping into a bunch of different systems in order to expose inventory information—components and stock, work-in-process, and finished goods—at every point in our supply chain. If we can't see it, we can't measure it and reduce it. I'd like to get your help to make that program happen. You need to talk with John about this, but let him know we've talked and you're to help drive this initiative." Sedgewick nodded. John Bentley was the Chief Operating Officer (COO) responsible for manufacturing, R&D, and overall operations for the company. Bentley was fairly receptive to new ideas if they could drive the metrics of the business, so Sedgewick was confident about defining and implementing the processes and systems to help better manage the company's inventory.

"Anyway, I wanted to pick up our discussion from lunch last week to see if there might be a way to use your ideas about Web services to help implement these initiatives. You were pretty bullish on the whole Web services thing, so let's continue with that for a bit." Dunston leaned back in his chair and looked expectantly at Sedgewick, waiting for him to begin.

Sedgewick decides to recap their previous discussion and then apply it to these newly articulated business strategies.

"Bob, you remember how I described what Web services are, right? The definition I like best is one I've synthesized from all the trade rags I've been reading. Here's my definition of Web services. Web services are 'loosely coupled, self-describing services that are accessed programmatically across a distributed network and exchange data [or information with one another] using vendor, platform and language-neutral protocols.' They are software modules or applications that are designed to be run across Intranets or the Internet using the underlying protocols that the World Wide Web is based upon today."

Dunston leans in with interest. "So, give me an example, Sedgewick."

"Well," Sedgewick replies, "think of when you said you wanted to see inventory information—let's say work-in-process inventory. In order to do that, you would need information from multiple locations from multiple Manufacturing Execution Systems (MES) and multiple Enterprise Resources Planning (ERP) systems, right?"

"Yes, and it's a pain in the neck to do that, right?" Dunston half asks and half states.

"It is and it isn't," replies Sedgewick. "These days, you would typically use tools such as Enterprise Application Integration (EAI) that specialize in tying disparate systems together. There are a bunch of products that do this today, and they vary in what they do and how they do it. They can be expensive, and require adapters or interfaces to tie into back-end systems and extract information in the manner in which it is desired. It depends on the business need that drives the use of the tool."

Sedgewick continues, "Now, let's suppose you want to get inventory information from the manufacturing execution systems and ERP systems in three different plants in three different geographies to update an inventory management portal." Sedgewick stands up and walks to the whiteboard, unwrapping a Snickers bar while he uncaps a dry erase marker. He draws a blue box to represent headquarters and three additional blue boxes to represent the international manufacturing sites. "We have a portal running here at headquarters, and we are populating the portal with real-time inventory updates from the plants. That means we have to gather information from the ERP system at each plant. SAP in this plant, JD Edwards for

this plant, and Oracle Applications for this plant. It's a pretty typical scenario for many organizations."

Sedgewick pauses, exchanging the blue marker for a red marker. He draws red lines from each of the blue plant boxes, connecting them to the headquarters box. "Now," Sedgewick continues, assuming the instructional tone of a college professor, "updating the headquarters inventory management portal can take place in a number of ways. For example, there could be a real-time connection from the portal application, over our internal network, tapping directly into each ERP system. Beyond that, we could add logic to push information from the ERP systems, only processing updates if there has been a change, versus pulling all the inventory information and updating the portal application regardless of whether it has changed or not."

"Solving this problem with EAI software is pretty typical, and it works. The only problem is that these point-to-point interfaces can be cumbersome to maintain, and EAI software can be expensive to purchase, install, and maintain."

Dunston's face wrinkles as the word "expensive" enters the conversation, but he doesn't say anything. Sedgewick, noting the change of expression, quickly responds. "That's why Web services are such an exciting and timely technology, Bob."

"Theoretically, Web services can eliminate the integration problem that we would use EAI software for, and it's ideally suited to loosely coupled interfaces between applications or business processes, much like the inventory portal problem we are discussing." Sedgwick walks to the whiteboard again. "You see, these red lines represent the connections we would build using the EAI software, and these processes would remain in place to pull inventory data from the targeted ERP systems on a periodic basis, either polling them or being updated as inventory information changes. EAI implementations are fine, but they have some limitations. There are platform and version issues to contend with, such as HP's Unix, IBM's Unix, and Sun's Unix versus Microsoft Windows NT, all versus Linux—the OpenSource version of Unix. Then, you have the application software itself and the task of making sure that the versions of software are the same, or at least can be accessed using the same EAI software and adapters. And, in many cases, the ERP software has been customized such that it really isn't the standard functionality the vendor originally offered. Web services offer a better way to make applications interoperate using Internet protocols and emerging Web services standards."

Exchanging the red marker for a green one, Sedgewick draws small green boxes inside each of the blue boxes on the board. "Now, with a Web services approach to this problem, we would build small, modular applications that perform simple computing tasks—for example, retrieving inventory updates

from ERP systems." Pointing to the little green boxes, Sedgewick continues. "Each of these little applications are Web services, written in Java or C/C++—it really doesn't matter. In addition to the main application functionality, which is simply to get inventory records from the ERP system or database, they have some eXtensible Markup Language (XML) code added to them."

At the mention of these acronyms, Dunston's face brightens because he has read about XML and Web services in the *Harvard Business Review*, so at least he understands some of it. Dunston quipped, "You mean, expensive markup language, don't you?"

"Yeah, right," chuckles Sedgewick as he proceeds. "The XML code that I am referring to does two simple things: It has a SOAP protocol layer, which is Simple Object Access Protocol. The SOAP protocol is a messaging standard to format messages between Web services consumers and producers. SOAP specifies the message envelope, the header and the message body—all in XML. This is how a consumer or user of a Web service and the producer of a Web service communicate via Internet protocols, typically HTTP."

Dunston furiously scribbles notes as Sedgewick describes these primary Web services standards for messaging. Sedgewick watches Dunston's facial expression as he continues, making sure that there is no confusion or boredom in his eyes. "The other piece of XML code that is added is known as Web Services Description Language (WSDL). This XML-based standard describes how a Web service is accessed and what its inputs and outputs are. WSDL provides the interface to the Web service so it can be used programmatically by other Web services without ever needing human intervention."

"What this all means," says Sedgewick, "is that these software components are designed to work together, over the Internet, or in this specific case over our corporate intranet, to gather inventory information from three different systems and aggregate it in the inventory portal. Using Web services in this way can help us integrate legacy information systems by exposing important business information using standards such as XML, SOAP, and WSDL—as opposed to often expensive EAI tools and proprietary and inflexible integration techniques."

"Will our integration expenses go down with Web services?" Dunston asks. His inquiry draws a quick grin from Sedgewick.

"It's conceivable that over time, the effort and expense associated with internal systems integration will be reduced significantly as Web services are used to expose information from proprietary business systems for use by other business applications. Web services could reduce or eliminate the need for EAI tools because Web services are based on standard Internet protocols and XML. As more and more Web services are made available,

initially within organizations and eventually publicly via shared Universal Description, Discovery, and Integration (UDDI) registries, the need for specialized integration software will decline."

Dunston leans back in his chair and folds his hands behind his head. "Okay, Bill, you've explained what Web services are—at a high level, I'm assuming—and you've said they can reduce the amount we spend integrating systems together within the organization. I understand that. But I can't imagine that's where all the benefits of Web services will be realized. There has to be more to it than that."

Sedgewick, almost anticipating this question, jumps back up to the whiteboard—again with a marker in hand. "You're right, Bob—there is."

"We've talked about doing some acquisitions to fill the R&D pipeline, right?"

Dunston nods, answering, "Yes, that's right."

"So think about the inventory update example I started with. You have these small, distributed applications that we can assemble together, via Web services standards and Internet protocols, to retrieve inventory information from proprietary ERP systems running on distinct hardware and operating systems. This entire process is running over our intranet."

"Okay, so say we buy a company. What's the first thing we have to do?"

"Integrate the financials," Dunston quickly replies. He had orchestrated multiple acquisitions during his career and knew the playbook by heart. "Integrate financials first, then rationalize and consolidate product families and customers into a single view of the business."

"Right," replies Sedgewick. "And one potential way to do that is to use Web services to tap into the acquired organization's financial reporting systems to aggregate their financials and report them back to us here at corporate."

Dunston nods vigorously. "But wouldn't that mean using Web services across the internet? Aren't there security issues still?"

Sedgewick responds with marker in hand, "Well, from an IT perspective, one of the first things we'd do is dismantle the acquired firm's firewall and bring them onto our private network as soon as the deal closes. So, really this is similar to the internal integration example. Nonetheless, it offers significant benefit in quickly integrating an acquired company into the operations of a parent company, regardless of the IT infrastructure or the application portfolio choices of the acquired firm."

"The interesting and perhaps most elegant part of this approach is repurposing of the Web services used for the M&A integration for other acquisitions and internal integration needs. That's a real benefit of Web services—the ability to reuse services because they are open, standards-based and flexible as opposed to being rigid, monolithic software applications with proprietary interfaces."

Sedgewick, looking pleased with himself, sits down across from Dunston and finally puts the marker down on the table as Dunston finished writing a few more notes in his legal pad. "So, based on our discussion, what would you suggest we do to get started here? Are there some things we should begin doing right away, as well as planning for the future?"

Sedgewick promptly replies. "First, I'd like to get some of my core team educated about Web services, and I also suggest we arrange a briefing for all of the executive team as well. That will be important so we're all speaking the same language about Web services and their potential. Next, we can begin prioritizing the list of business and technology initiatives you're considering for next year and see where we can drive their completion faster with Web services and realize the business benefits more quickly."

Dunston tilts his head back and looks up at the ceiling. "What about the M&A integration issue? I'd like to begin working on that process as well—perhaps putting some kind of a specification or architecture together describing how we can use Web services to shorten the integration time of an acquisition." Sedgewick nodded his agreement.

"Okay, we'll put together a briefing describing how we might streamline the systems integration component of the M&A process, what we would need to begin developing a Web services framework to achieve it, and when it can be tested and ready. We can fine-tune it once we begin the due diligence process."

Satisfied, Dunston finished taking notes. "Good. Let's fast-track the executive briefing and get your team trained, and I'll have a standing agenda item every two weeks at our staff meetings to discuss how these Web services initiatives are progressing. Let me know what resources you need."

The meeting ended, and Sedgewick exited Dunston's office—bidding him goodbye as he walked out. Both executives are thinking the same thought as they part: "Web services are going to have a significant impact on the way we do business! We'd better make sure that we're ready to take the Web services initiative before our competitors do."

Both Dunston and Sedgewick are correct. Web services will have a significant impact on the way they do business in the coming years. Arising from the dust of the dot-com boom and bust, Web services are perhaps what the Internet should have been originally. While we are still studying the business lessons of the first Internet wave, one thing is clear: Technology does not stand still, even though business conditions might be difficult. Once the technological genie escapes from the bottle, the possibilities that are unleashed are endless.

Web services will change the way that organizations locate, research, assemble, test, and deploy software to solve business problems, as well as

how they tackle new market opportunities. Web services will have a profound impact on the way in which software companies build, sell, and deliver software to their customers. Web services will enable the traditional IT organization to truly evolve into a strategic business asset, no longer relegated to the status of a support organization. The corporate computing model will change, and the management skills required to navigate these changes will be as much business and strategy as they are technology.

A BRIEF RECAP: THE PAST 10 YEARS

During the dot-com bust of the past two years, something amazing occurred. While the business world watched dot-com after dot-com fold, and the luster of e-Business and the Internet steadily dulled, the technology visionaries were hard at work. The business world was under attack on a number of fronts, including the economic slump, the technology hangover from e-Business spending, and rapidly changing business conditions. Add the economic recession and the financial scrutiny caused by Enron's stunning collapse, as well as the failures of venerable organizations such as K-Mart and Global Crossing, and you have the makings of trouble.

The business issues facing organizations over the past few years have been considerable. Between year 2000 preparations, then the tremendous investment in e-Business followed by the economic slump of the past two years, organizations have faced travail upon travail. Of course, the world of IT has also been pressed. Organizations have invested millions in ERP platforms, CRM solutions, and e-Business initiatives. They have implemented the supporting applications and infrastructure to drive their businesses via the Web, such as content management, enterprise portals, data warehousing, and analytics solutions. Organizations have spent hundreds of millions of dollars on complex IT solutions—solutions that often resulted in application silos that are massively inflexible and extremely difficult and costly to implement. Many organizations spent tens of millions of dollars and several years installing ERP solutions, only to find that by the time they were through, their business had changed—both internally as well as externally. The monolithic application footprint and rigid architectures of ERP solutions have created a host of copycat organizations that have implemented the same solutions in the same industries. This resulted in lost competitive advantage and uniqueness of business models. It resulted in an inability to change business processes as well as information systems to meet emerging needs. ERP implementations resulted in a business architecture based on internal operations versus interactions with customers and suppliers. The difficulty of integrating ERP-centric backbones with other organizations' ERP backbones persists and has created a

substantial market for Business to Business (B2B) integration tools such as Enterprise Application Integration (EAI), messaging software, and other middleware solutions.

The Internet exposed this problem even more as organizations attempted to link their businesses and systems in support of new initiatives such as collaboration, partner relationship management, product life cycle management, and other emerging business needs. Business strategies such as M&As drove a need to rapidly integrate an acquired organization's business processes and IT systems. Issues such as master data management, managing customer and product information, and eliminating redundant systems have plagued businesses in M&A mode. Productivity gains were sought through initiatives such as employee self-service and enterprise portals, which made it easier to access relevant information by role and needs. B2B collaboration has to date been inhibited by expensive integration efforts, inflexible enterprise business applications, and the inability to extend and/or augment existing business applications to accomplish new business functions as business needs change.

The Enterprise Application Phenomenon

The enterprise application phenomenon is the culmination of the client-server era of computing. Client-server computing based on the three-tier, then the n-tier architecture, completely revolutionized computing as it was known in the early 1990s. Client-server computing models broke the paradigm of large centralized mainframes serving masses of dumb terminals based on the dramatic rise of the Personal Computer (PC) in corporate and home computing. Client-server computing hailed the introduction of the distributed computing model, where applications could be built and deployed more efficiently and targeted to distinct business audiences, departments, and end users, as opposed to being driven by internal IT organizations that were the traditional buyers of computer applications. The client-server architecture created new "markets" for software vendors to sell to by virtue of the increasing development of departmental and functional applications that solved targeted business problems, yet interacted with other functional applications or modules from the same vendors. SAP R/3 arguably represents the pinnacle of client-server success. This ERP solution is the industry's leading business computing platform, followed by offerings from Oracle, PeopleSoft, and a host of others. SAP's dominance in the ERP arena can be explained by a number of factors, but one surely is that it was among the first business application suites to be built on a client-server architecture. When SAP R/2, the mainframe version, was replaced by its R/3 client-server release, SAP took off and never looked back. SAP rolled over all the competitors in the ERP space and established

itself as a major force. The Big 5 consulting firms were profiting handsomely from SAP implementation services, and analyst organizations were raking in fees for research, vendor selection, and analysis. The SAP ecosystem was rich with revenue opportunities that supported or complemented the SAP ERP solution. PeopleSoft, which was also known for its client-server architecture in the human resources arena, attempted to challenge SAP by acquiring other vendors and rapidly expanding its application footprint to additional functions and departments of the business enterprise. SAP had an advantage in functional breadth, however, and the battle was won before it began.

The point of this story is that client-server computing brought with it a number of changes in the way that applications were developed, sold, and implemented by software vendors and consumed by corporate users. The client-server architecture created a wave of change across the entire information technology value chain, threatening the incumbents and embracing new entrants who had a new way, a better way, for computing to be performed. The PC was clearly one of the drivers of the client-server wave, as well as the desire to break the traditional highly centralized mainframe computing model. The client-server computing paradigm for the first time invited end users into the corporate computing dialog, and extended the reach of business software from the IT department to all business departments. Client-server computing, based on the widespread penetration of PCs into homes around the world, enabled users to do more with technology.

Today, Web services are about to create a new wave of change. This new engine of change is relatively simple: The catalyst for Web services is agreement. At its foundation is agreement on the adoption of three fundamental standards for communicating between computer systems: TCP/IP, HTTP, and XML. TCP/IP, or Transmission Control Protocol/Internet Protocol, Hypertext Transport Protocol, and Extensible Markup Language are the pervasive standards for computing that emerged from the Internet revolution. The Internet capped the client-server era of computing by making computing pervasive to all users, in organizations, and in homes. The Internet extended the reach of computing into virtually all aspects of the human experience, from CEOs of multinational conglomerates to home users e-mailing vacation pictures to relatives. Much as the Internet broke the communication and information bottleneck for information consumers of the client-server computing model, Web services will break the communication and information bottleneck for business enterprises.

Web services will take B2B communication to new levels. The Internet enabled personal collaboration via e-mail and instant messaging tools; Web services will enable corporate collaboration via loosely coupled applications across organizational boundaries. Web services will enable much

more than information exchange between organizations based on dedicated interfaces at the system level. Web services will enable businesses to interoperate at the business process level in dynamic and emergent ways as new processes arise in response to changing business conditions and changing corporate priorities.

Rise of the Wintel Duopoly

While client-server computing rode the PC-driven Wintel (Microsoft Windows operating system and Intel microprocessor) wave, the Internet began exposing this architectural paradigm's weaknesses. Web browsers removed application-specific user interfaces as the method of choice for navigating applications and content, and the rapid adoption of the Web meant that desktop computing wasn't really what users wanted. PCs were not used as computing devices; they were used as communication devices. As mobile computing has increased in popularity, and as the inherent difficulties in business-to-business integration have been realized, a new paradigm of computing is being hailed as the solution. Web services are here.

The inevitable saturation of homes with PCs, combined with the rapid rise of wireless devices, has huge implications for Microsoft and for Intel. First, slowing license revenue from Windows and related desktop software is forcing Microsoft into new ways of revenue and profit creation, such as set top boxes and gaming devices. Microsoft's core business of desktop operating systems and desktop software will be increasingly threatened as computing devices move away from the desk and become increasingly mobile. Slowing sales of perpetual software licenses has Microsoft concocting new ways to drive revenue growth, and one of those ways spells the end to these licensing arrangements. Some see a future in which software is sold as services through subscription fees. Microsoft is already considering a rent-for-use model for Microsoft Office, much like subscribing to cellular service and cable television. ERP vendors might consider licensing modules on a metered, per-click basis as opposed to licensing on a per seat basis. Some speculate that Web services will help to revive the flagging Application Service Provider (ASP) market and perhaps make business applications more affordable for all organizations, especially those in the mid-market that can not expend the millions of dollars required for a typical ERP implementation.

Web services are perceived by some skeptics as an attempt by Microsoft and other large enterprise software platform vendors to halt eroding license revenues by providing software as services. This action will help stave off declining revenue from slowing PC sales and the shift from desktop computing toward mobile computing. Others see IBM's Web services

thrust as a way for it to maintain hardware and services revenue by positioning itself as the software and platform vendor of choice, much as Sun Microsystems was perceived as the Internet platform of choice.

While there might be some truth in these ulterior motives by some platform vendors, Web services are farther reaching than that—and the benefits are far too compelling to ignore. The fact that all major software vendors have embraced the standards of Web services, and are racing to develop tools and solutions to facilitate the adoption of Web services, shows how the move toward Web services is beyond the span of control of any single software vendor.

BUSINESS VALUE FROM WEB SERVICES

Web services will drive new levels of collaboration between companies in existing value chains as well as enable new relationships with trading partners in emerging value chains. This situation will occur because of the friction-reducing promise of Web services, making it easier to perform B2B integration at the business process level. The last several years have witnessed the rise of middleware solutions to solve the problems of tying business systems together, to perform transactions and information exchange across organizational boundaries. Enterprise Application Integration (EAI) tools emerged, messaging-oriented middleware took hold, and a host of similar solutions addressed the need to make application portfolios work together within the organization and across organizational boundaries. Middleware solutions, and now Web services, present the opportunity to solve a number of broad business and technology issues, including the following:

- Reintroduce the business flexibility that Enterprise Resources Planning (ERP) and other large, enterprise applications removed through rigid business process definition and proprietary application interfaces.
- End the debate about IT alignment by allowing the idea of Just-In-Time (JIT), or the implementation of new applications as the business needs them without the implementation and integration lag that accompanies large, enterprise software implementations.
- Extend CRM, ERP, and other large, monolithic software applications to add new business functions or capabilities in response to changing business needs.
- Provide connections to other trading partners for collaborative processes such as forecasting and supply chain planning, transaction management, and others.
- Help organizations manage change given their existing reliance on large, legacy systems and change-resistant business processes.

- Revolutionize the corporate computing model for software-producing companies, for software-consuming companies, for hardware and infrastructure companies, and for services companies.

Specifically, Web services promise to deliver the following high-level benefits to organizations:

- Support application integration internally and externally, across the firewall.
- Provide easier B2B collaboration between business partners.
- Transition software deployment models from big-bang, high-footprint implementations to just-in-time applications that are appropriate to the business challenge being addressed. This is the beginning of the idea of Just-in-Time Information Technology based on Web services. Software rental models and grid computing are simply specific subsets of the idea of JIT IT.
- Reduce costs of software procurement, deployment, and integration.
- Increase business and technology agility by deploying loosely coupled business applications versus monolithic enterprise applications.

Why Was the Internet Not Sufficient?

One question about the Internet is why it was not robust enough for true B2B collaboration, as well as for complex processes that span multiple Web sites or multiple organizations for Web transactions. The Internet rapidly evolved for B2B transactions, yet it fell short for complex collaboration between organizations due to the following issues:

- Lack of standards for B2B integration and B2B automation
- Manual searching of Web content by browsers versus registry-based applications that find one another and auto-invoke (this result is a future but possible scenario)
- Emerging standards such as XML have already become fragmented, diminishing their ability to create consensus among industry solution providers and consumers
- Internally-focused application procurement and deployment models

Of all these, the most difficult to overcome is probably the internal IT architectural models of organizations. Internally focused application procurement and deployment models have to date dominated the IT architecture of organizations, in contrast to collaboration-centric application models. Most IT architectures of today's corporations are based on a single-company view of their own internal operations. That is not necessarily bad or wrong. It merely means that an organization's investment in its IT

architecture—its infrastructure, transaction systems, and application portfolio—has been focused on internal operations and efficiencies, not on interacting with trading partners and customers. In various industry value chains, for example, the application portfolio of the dominant firm in the value chain exerts a strong influence on how its trading partners interact, both with the dominant company as well as among the trading partners. This dominant application influence often forces trading partners to embrace the nuances of that architecture and the interfaces to the ERP applications. In an SAP-centric extended enterprise, trading partners will have to interface to SAP to exchange forecasts, purchase orders, and other B2B transactions.

However, in today's business world, collaboration with trading partners is fast becoming the rule, not the exception. Companies understand that significant benefits can be realized through better cooperation and information sharing with their customers and suppliers. Their existing application portfolios, however, are not built for collaboration across the firewall with outside agencies. Web services offer a way to bridge the gap and overcome the legacy of internally-focused IT architectures and application portfolios.

Business Process Collaboration

Web services will enable business collaboration at the process level. Process-level collaboration requires new software architected for collaboration across corporate firewalls. Web services will be the foundation for creating these new applications. Business Process Collaboration (BPC), augmented by electronic means over the Internet, has wide-reaching implications for the ways in which business will be performed. Many organizations have not had the discipline or desire to focus on business processes as a legitimate pursuit, largely as a result of the ongoing backlash against the business process re-engineering phenomenon of the 1980s. However, as the word "collaboration" has entered the mainstream dialog of business and IT professionals, the sharp edge of re-engineering has been dulled. Collaboration as a discipline is on the rise as the Internet continues to thread its way into organizations around the world.

We can simply define collaboration as cooperation to achieve a particular goal or goals. Collaboration involves teaming, sometimes with competitors, to achieve a higher, shared purpose. This is sometimes called co-opetition, which refers to the periodic vacillation between competing with organizations and cooperating with organizations based on market dynamics, competitive pressures, or other business forces. Collaboration has been around in various forms for many years, including incipient technology implementations such as Electronic Data Interchange (EDI), point-to-point interfaces between application systems, and other means. What is different is that the Internet has

opened the door for organizations to exchange information electronically over a set of agreed-upon standards, collectively known as the Internet standards of TCP/IP, HTTP, and XML. The Internet altered the cost structure of collaboration for organizations because of the global acceptance of these technology standards. EDI has been around for years, for the most part implemented by large corporations that could afford to install the technology for inter-organization purchasing processes, forecast sharing, payment and reconciliation of shipments, and more. The Web has changed the cost equation for exchanging business information between firms, which means the price of entry is now much lower. A rich variety of new solutions are embracing collaboration, or c-commerce as some analysts have dubbed it. Early implementations of collaboration demonstrated the potential—yet they also illuminated the shortcomings of existing Web technologies and standards to support this rapidly emerging space. But the act of collaborating with another organization, not to mention many organizations, requires new ways of connecting businesses together. Collaboration is far more than the static exchange of e-mail or spreadsheets. It is interactive. It is live, or real time (or should be as required by the business process being driven by collaboration). Business process collaboration necessitates more than limited information exchange.

The brutal shakeout of dot-coms, further complicated by the economic recession, proved that despite all the glamour of e-Business and technology, there was an obvious problem. The business environment had not really changed for most businesses, and the rules of survival had not changed either. Business models were still stagnant, fixed, and unchanging. IT architectures are still ERP-centric, prone to inflexibility, and very difficult to customize and extend to support emerging business needs. These application strategies lock an organization into a fixed way of conducting internal operations according to the business model of the software vendors. Implementing a typical ERP application requires the adoption of a view of internal operations that substantially complies with that of the software vendor and its other customers. This situation removes opportunities for unique competitive advantage through business process execution and operational excellence.

ERP-centric architectures are based on the architectural philosophy of internal operations efficiency. Building a single view of internal operations based around a centralized database of the organization allows visibility of financial metrics, inventory levels, customer information, orders, and more, all in real-time. This capability is powerful, despite some of the issues that attend these implementations. These issues include, but are not limited to, the following:

- Rigid architectures freeze corporate operations into a fixed, inflexible model
- Complex architecture means changing one module often necessitates changes in many other modules

- Large software footprint and module inflexibility result in long, expensive implementation cycles that, when completed, are already outdated
- Internal orientation and architecture of ERP systems limits the ability to engage in collaboration with outside trading partners. This situation is primarily due to proprietary application interfaces and business process inflexibility forced onto a business by the ERP architecture

Streamlining internal operations and allowing financial management of an organization's operations was a significant benefit of ERP systems. However, times have changed for all firms. Inward-focused systems and business processes can only deliver limited value in a world dominated by a desire, and more importantly a mandate, to work with other trading partners to accomplish business success. The rapid rise of the Internet exposed the inadequacy of ERP and other enterprise systems to rapidly accommodate new business processes. As new business needs continued to grow, the increased need for agility and new business functionality has outstripped the ERP capabilities of today.

There has been a dramatic shift from the internal focus of ERP systems to collaboration with outside trading partners. This shift from internal operations to collaborative interaction with external trading partners has challenged the business processes and the IT application portfolio of most organizations. The shift to the front for CRM and other e-Business applications has placed an unprecedented demand on IT systems, and the need to securely share internal, potentially sensitive, information will continue to increase. However, as these inter-enterprise collaboration requirements continued to grow, the technology supporting collaboration proliferated. Enterprise application integration and other collaboration products emerged as the next hot space. EAI continues to be an important area, and will serve as one of the critical pathways to Web services because these tools are designed for connecting systems and enterprises across the firewalls.

As you will see from the Web services adoption model—introduced in the following section—pragmatic uses of Web services will carry us for the foreseeable future. As the standards and technology progress, more sophisticated Web services capabilities will emerge.

WEB SERVICES ADOPTION MODEL

Web services, like the Internet during the mid-1990s, will be adopted in phases based on what a company hopes to achieve and how it desires to operate within its chosen markets. It is important to note that there are significant differences that distinguish the adoption of Web services when compared to the rise of the Internet. Firstly, there is widespread

agreement on the basic standards of Web services. All major platform providers, software vendors, and professional services organizations are embracing Web services standards. Secondly, Web services hold great promise for the realistic support of complex B2B transactions and processes that span organizational and business boundaries. Web services will be able to fulfill these capabilities primarily due to the broad support within the software industry for the acceptance of Web services standards.

Web services will be adopted in four distinct phases. These phases are based on how Web services will be implemented within organizations, within industries, and across the global business landscape. They are based on how businesses evolve and absorb new capabilities rather than on the use of technology for technology's sake. Web services will initially deliver business value through enablement of information integration and collaboration, followed by increased innovation as new uses of Web services are devised. Finally, the effective use of Web services to enable superior business execution will lead to the separation of market leaders and first movers from the rest of the pack.

The four phases of Web services adoption are:

1. Integration
2. Collaboration
3. Innovation
4. Domination

These four phases of change capture how organizations will enter into the world of Web services conservatively at first with internal integration projects, and then expanding into inter-organization, cross-firewall implementations with trusted partners, followed by further-reaching implementations with a network of trading partners. As cycles of learning are executed by the first movers, they will begin a wave of rapid innovation with Web services, developing new, industry-shaping and market-making distributed business solutions. These solutions will have the potential to completely reshape the competitive landscape of an industry. As innovation continues, and competitive advantage is extended through faster execution of industry-leading capabilities and strategic execution, these Web services thought leaders will eventually dominate their respective industries. They will demonstrate first-mover advantage. They will be the leading edge organizations that understand the information imperative of competition today—that information-based business models are critical to winning in today's global economy.

Figure 1.1 depicts the evolution of Web services according to the Web Services adoption model.

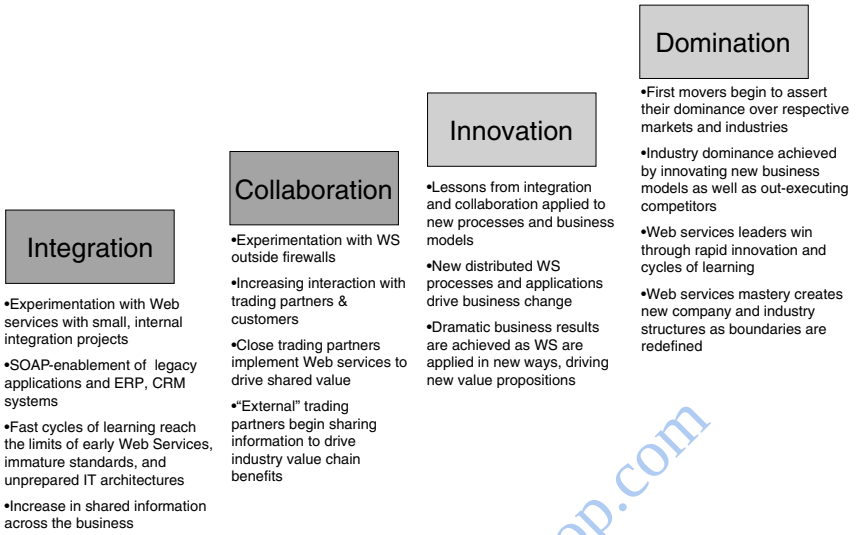

FIGURE 1.1 Phases of Web services adoption.

Phase 1: Integration

The first phase of Web services adoption will begin with internal system integration projects. The need for internal integration derives from the myriad of information silos created by proprietary enterprise applications implemented to support activities such as financial management (general ledger, accounts payable, accounts receivable), costing systems, order management, procurement, and production scheduling. These enterprise applications are typically large, client-server implementations built with an internal, organization-facing view of the world, and oriented toward internal efficiency and controls.

In their initial deployments of Web services, organizations will write services that expose the functionality locked within enterprise applications and legacy systems, enabling that functionality to be leveraged by other applications or business processes. The integration phase of Web services adoption will prepare organizations for the next phase, collaboration. The lessons learned from applying Web services internally to systems integration and enterprise application integration problems will be leveraged for the benefit of external trading partners in the collaboration phase of Web services adoption.

Phase 2: Collaboration

The collaboration phase of Web services adoption will drive process and operational improvements in many business areas, provided that the integration hurdles can be overcome and that the tools and technologies are mature enough to enable true collaborative behavior. Maturation in the collaboration phase of Web services adoption will prepare organizations for the next phase, innovation. This phase of Web services adoption will spur a new wave of rapid business change, which will largely shape the next wave of Internet expansion worldwide.

Phase 3: Innovation

During the innovation phase of Web services adoption, organizations will devise completely new ways of doing business based around Web services. These firms will leverage what has been learned from internal integration projects and from collaboration projects with outside customers, partners, and suppliers. These organizations will be able to turn these lessons into new business processes and new sources of competitive advantage. They will use Web services as an innovation platform to drive new levels of business performance along multiple dimensions of their value chains.

The innovation phase will spur a wave of new ideas for how business processes can be distributed, organized, and executed across corporate and industry boundaries. With Web services, the notion of industry convergence has a higher probability of realization because of the widespread agreement that it is the right way to conduct business. The innovation phase of Web services adoption will eliminate major roadblocks to the widespread use of Web services to drive new business process innovation and, ultimately, dramatic levels of business execution and performance.

Phase 4: Domination

The domination phase of Web services adoption will be the culmination of the previous three phases: integration, collaboration, and innovation. The domination phase is where the winners are separated from the also-rans, based on their ability to drive superior business value through the use of Web services in Distributed Business Process Execution (DBPE).

The domination phase will be based on superior performance in business as well as in the use of and innovation in Web services. Dominance will be established by a few organizations in each industry that realized the potential of Web services, both in changing internally the ways in which organizations can operate and outperform their competition using their information technology capabilities.

This introduction to the Web services adoption model shows a very real high-level scenario for how Web services will be deployed by corporations. Chapter 3 explores Web servcies adoption in greater detail.

SUMMARY

Web services are on the rise for many reasons. Some of these reasons are good for businesses, some are good for consumers, some are opportunities for entrepreneurs to create new business paradigms. It is not our intent to over-hype Web services with reckless exuberance. Nor is it our intent to underplay just how this rapidly emerging set of business ideas and technologies might impact our daily lives. We seek the middle ground while striking a compelling tone.

The value chain of information technology promises to shift radically with the emergence of Web services. The emergence of Web services has already created many new companies hoping to tap into this new paradigm, and there will be many more. By the time readers have digested Chapter 8 of this book, "The Web Services Vendor Landscape," which is a brief vendor survey, some of those companies will either have been acquired or will have gone out of business. Welcome to the Darwinian world of emerging technology.

Web services will impact the business world in new, unexpected ways. But before you, as a business executive or IT professional, make up your mind about Web services too quickly, remember one thing: Business benefit will carry the day for Web services, not technology alone. Read on, and think *business value!*