

zapthink white paper

THE BUSINESS CASE FOR WEB SERVICES



THE BUSINESS CASE FOR WEB SERVICES

March 2003

Analyst: Jason Bloomberg

Abstract

Today's IT executive doesn't have the time or the budget to invest in new technologies that fail to offer rapid business value. It is important, therefore, for managers to understand both the short-term and long-term value of Web Services. They must be able to cut through the hype surrounding this new approach to computing and understand how their organizations can use Web Services today to reduce the cost of integration in the enterprise. Even more important, however, they must understand the strategic value of Web Services.

The strategic value that Web Services offer is in the form of business agility: IT infrastructures flexible enough to enable companies to leverage changing business environments for competitive advantage. To build such agile infrastructures, organizations must build Service-oriented architectures, leveraging Web Services strategically across their IT environment.

Service-oriented architectures, while easy to conceive in theory, are difficult to build in practice. Companies should take a step-by-step approach that requires both a new perspective on IT architecture as well as a Web Services management solution that enables the construction of Service-oriented architectures.

All Contents Copyright © 2003 ZapThink, LLC. All rights reserved. The information contained herein has been obtained from sources believed to be reliable. ZapThink disclaims all warranties as to the accuracy, completeness or adequacy of such information. ZapThink shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve its intended results. The opinions expressed herein are subject to change without notice. All trademarks, service marks, and trade names are trademarked by their respective owners and ZapThink makes no claims to these names.



Table of Contents

I.	Where is the Business Value in Web Services?	4
	What are Web Services?	4
	Web Services quick win: reduced costs	5
	The business motivation for strategic Web Services	6
	Business agility: the strategic business benefit	6
II.	SOAs – the Key to Web Services’ Business Value	7
	Crossing the chasm from individual Web Services	8
	Web Service networks: putting the pieces in place in the enterprise	9
	Service orchestration: providing value to business users	10
	Business process intelligence (BPI): closing the agility loop	11
	Service orientation: the next major enterprise computing movement.....	11
III.	Digital Evolution: Comprehensive Web Services Management	11
	Single Service management.....	12
	Managing Service networks	12
	Orchestrating Web Services	13
	Providing business intelligence	14
IV.	Conclusion: the Need for Comprehensive Web Services Management.....	14

I. Where is the Business Value in Web Services?

Today's enterprises have little time for new technology advancements that don't clearly meet the needs of the business. IT budgets are tight, and many indications imply that this trend is not going to loosen up much very soon. Nevertheless, many enterprises are still struggling with a range of difficult problems resulting from years of implementing technologies in heterogeneous environments, the accumulation of multiple, complex integration projects, IT infrastructures that are too brittle, and an ever increasing litany of challenges. In this environment, an evolutionary approach to distributed computing based on standards has emerged under the *Web Services* name that is intended to address many of the issues facing IT organizations today. There's a lot of noise surrounding Web Services, to be sure, but the questions facing IT leaders in today's enterprises are: just what are Web Services, and how can they help solve the real IT problems we have today?

What are Web Services?

A *software service* is a software component that exposes its functionality on the network so that other software can access that functionality. *Web Services* are software services that are *encapsulated*, *loosely coupled*, and offered via *standard protocols*. Essentially, Web Services are application functionality residing on systems that accept requests from other systems locally or across the Internet by means of lightweight, vendor-neutral communications technologies. Specifics of this definition include:

- *Encapsulated* means that the implementation of each Web Service is invisible from outside the Web Service. Its functionality is known only by the interface it exposes. In essence, Web Services abstract the underlying implementation from the interface.
- *Loosely coupled* means that Web Services and the programs that invoke them (known as Web Service consumers) can be created and changed independently of each other, without requiring a redesign of the involved components.
- Web Services are typically built upon the standard protocols XML (eXtensible Markup Language) and HTTP (Hypertext Transfer Protocol), which are both open and freely available. In addition, Web Services leverage SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Description, Discovery, and

Web Services are application functionality residing on systems that accept requests from other systems locally or across the Internet by means of lightweight, vendor-neutral communications technologies.

TAKE CREDIT FOR READING ZAPTHINK RESEARCH!



Thank you for reading ZapThink research! ZapThink is an IT market intelligence firm that provides trusted advice and critical insight into XML, Web Services, and Service Orientation. We provide our target audience of IT vendors, service providers and end-users a clear roadmap for standards-based, loosely coupled distributed computing – a vision of IT meeting the needs of the agile business.

Earn rewards for reading ZapThink research! Visit www.zapthink.com/credit and enter the code DIGBIZ. We'll reward you with ZapCredits that you can use to obtain free research, ZapGear, and more!

For more information about ZapThink products and services, please call us at +1-781-207-0203, or drop us an email at info@zapthink.com.

Integration), which are all standard protocols based upon XML.

Web Services are only the trees; there is still a forest that many people are missing—the over-arching vision of Service orientation that gives Web Services strategic business value.

The definition of Web Services suggests that they serve an important role in reducing the cost and complexity of integrating systems, especially if the systems are heterogeneous. As this paper will show, however, simplifying integration is only part of the value that Web Services can offer the enterprise. The true, strategic win for companies that implement Web Services comes when they apply the standards-based, loosely coupled principles of Web Services to their entire IT environments. The point this paper makes is that Web Services are only the trees; there is still a forest that many people are missing—the over-arching vision that gives Web Services strategic business value. That vision is *Service orientation*. *Service-oriented architectures* (SOAs) are an approach to designing distributed computing infrastructures that considers software resources as Services available and discoverable on a network. Open, standards-based Service-oriented architectures built with Web Services form a flexible approach to IT that provides business agility to the enterprise.

Web Services quick win: reduced costs

In today's enterprise, IT is typically the limiting factor that impedes an organization's ability to quickly make critical business decisions and changes. IT always seems to be too expensive, risky, or slow to respond quickly to changing business requirements. Of all the IT issues facing companies today, the one that is most likely to be a cause for these bottlenecks is *integration*. The seemingly simple act of getting two or more different systems to talk to each other in a flexible way that allows for cost-effective change has been a surprisingly persistent problem since the advent of distributed computing over thirty years ago. Some companies report anywhere from 40% to 70% of their IT budgets are spent on integration alone. Clearly, reducing the cost and complexity of integration will reduce many of the bottlenecks that constrain business.

For information technology to be able to deliver business agility, integration among systems must be rapid and cost-effective.

For information technology to be able to deliver business agility, therefore, integration among systems must be rapid and cost-effective. Why, then, are the problems with integration still troubling companies, even though distributed computing has been around for a generation or more? On one level, the cause is the lack of standard ways of programming different systems to communicate. For any two different systems, the traditional approach to integration is to write programming code for each system that teaches it how to talk to the other system. Such an approach is expensive and time consuming, and doesn't scale well or respond to change in a flexible way. This approach to integration is *tightly coupled*, which means that one programming team must control the integration code on both systems to get them to communicate with each other. Such integration is also point-to-point, which means that the complexity of the distributed systems explodes as the number of systems goes up.

Web Services address the issue of tightly coupled integration. By encapsulating disparate systems with Web Services interfaces, it's possible to dramatically reduce the cost of integrating those systems—some companies report savings of up to 80% over traditional forms of integration like Enterprise Application Integration (EAI). Using Web Services to simplify integration within the enterprise is technically straightforward and relatively low risk, thus offering enterprises a "quick win" that provides substantial cost savings with minimal investment in technology or human resources. This simple, point-to-point application of Web Services, however, only scratches the surface of the potential business benefits of SOAs.

While point-to-point applications of Web Services provide cost savings, they do little to reduce the complexity and brittleness of the overall distributed computing environment.

Instead of massive IT investment, today's IT executive is concerned with thrift.

Reworking existing brittle, expensive IT infrastructures into flexible, Service-oriented environments promises substantial cost savings most dramatically in terms of business agility: the ability to respond quickly and efficiently to changes in the business environment, and to leverage those changes for competitive advantage.

The business motivation for strategic Web Services

While point-to-point applications of Web Services provide cost savings, they do little to reduce the complexity and brittleness of the overall distributed computing environment. For enterprises to build agile IT infrastructures that reduce this brittleness, they must leverage the advantages of Web Services at the architectural level by building SOAs. In fact, business value of SOAs is so dramatic that enterprises across the world are considering how to transition their existing IT infrastructures to SOAs. This transition to Service orientation, however, is fundamentally different from the last distributed computing transition: the one from client/server to n-tier/Web architectures in 1996-97. That last build-out heralded the beginning of the dot-com boom, where Internet-related investment coupled with Y2K expenditures created a kind of IT “perfect storm” so dramatic it led to a worldwide economic boom, and subsequent downturn. Today, of course, the economic environment for technology adoption has completely changed, and this return to the “new business normal” is accelerating the move to Service orientation. Rather than promoting massive build-out or extensive rip-and-replace, Service orientation embraces heterogeneity and obtaining greater value from existing legacy technology. Today's distributed computing transition has an entirely different economic model from the last transition. Instead of massive IT investment, today's IT executive is concerned with *thrift*.

The Web Services story, in fact, is all about thrift. Point-to-point application of Web Services technology leads to substantial cost savings on integration. Thrift, however, means more than simple cost savings. True thriftiness means making do with what you have—squeezing value out of every asset. One of the benefits of an SOA is that such architectures help companies get more value out of existing resources by wrapping legacy applications in Web Services interfaces and then making those Services available on the network. A second thrift benefit that SOAs provide is that they facilitate heterogeneous IT environments. Instead of “ripping and replacing” existing corporate IT systems by installing new systems and throwing the old ones out, SOAs enable users to build bridges between different systems and applications and leverage existing IT assets.

Business agility: the strategic business benefit

Reworking existing brittle, expensive IT infrastructures into flexible, Service-oriented environments promises substantial cost savings, not just in terms of reduced integration expense and squeezing more value out of existing IT investments, but most dramatically in terms of *business agility*: the ability to respond quickly and efficiently to changes in the business environment, and to leverage those changes for competitive advantage. Change comes in many forms: changes in the marketplace, in technology, in the world at large. Companies that can make effective use of a changing environment are better able to compete and thrive in any business climate, but especially in tough economic times like those we have had since the bursting of the dot-com bubble.

Service orientation, however, has the potential to change the typical challenges associated with brittle architectures and IT bottlenecks, and enable business requirements to finally drive their technology decisions. On the other hand, building Service-oriented infrastructures is not easy. It requires investment and commitment on the part of enterprises. The long-term business benefits of Service orientation, however, can justify such investments. Therefore, the true goal of Service orientation—and hence, of Web Services in general—is to remove the bottleneck that IT has on businesses' ability to be agile.

II. SOAs – the Key to Web Services’ Business Value

There comes a time in the development of any market where competing vendors come together to hammer out common standards that provide for interoperability among each vendor’s products. Proprietary offerings soon give way to standards-based products, and vendors who cling to their existing product lines soon find themselves out in the cold, as customers increasingly demand interoperability and vendor independence. Whether it was alternating current, railroad track gauges, or dimensional lumber, the widespread adoption of standards led to market efficiencies, dramatically improved products, and substantial increases in customer value.

This movement to standards is now taking place in the IT industry as companies adopt Web Services and use them to build Service-oriented architectures (SOAs). Loosely coupled, standards-based SOAs encapsulate the functionality of the underlying software implementations, and present that functionality in terms of business concepts. Rather than focusing on the specific technical interface requirements of end systems, those Web Services that offer business-oriented functionality are *coarse grained*, which means that they offer their functionality in blocks of relevant information, instead of small pieces of data. Coarse granularity is one of the most important features of SOAs, as important as loose coupling and open standards.

In general, SOAs offer the following advantages over traditional approaches to distributed computing:

- They offer coarse-grained business Services, as opposed to fine-grained software-oriented function calls.
- They provide location independence: Services need not be associated with a particular system on a particular network.
- The Service consumers are loosely coupled from the underlying systems.
- Authentication and authorization of Service consumers, and in general all security functionality that applies to the Services, is also available via Web Services.
- Web Services consumers can find and connect to available Services dynamically.

SOAs, however, offer more than technical advantages over other approaches to distributed computing. Fundamentally, SOAs offer a different *perspective* on the role IT plays in an organization. Companies must understand this new perspective in order to get the full advantage of their Web Services implementations, and build truly agile IT infrastructures. To meet the needs of the agile enterprise, therefore, SOAs offer the following core principles:

- *The business drives the Services, and the Services drive the technology* – In essence, Services act as a layer of abstraction between the business and the technology. IT personnel must understand the dynamic relationships between the needs of the business and the available Services on the one hand, as well as the technical underpinnings that offer the layer of abstraction required by the Services on the other.
- *Business agility is the fundamental business requirement* – Instead of dealing with rigidly-defined, concrete requirements from business, SOA provides the ability to respond to changing requirements, which is the new “meta-requirement” for business. The entire architecture—from the

SOAs offer a different perspective on the role IT plays in an organization.

Companies should take a step-by-step approach to achieving Service orientation, first by creating and exposing individual Services, and then by combining them into a Web Services network.

hardware on up—must reflect the business agility requirement, because any bottleneck in an SOA implementation can substantially reduce the flexibility of the entire IT environment, and hence the business as well.

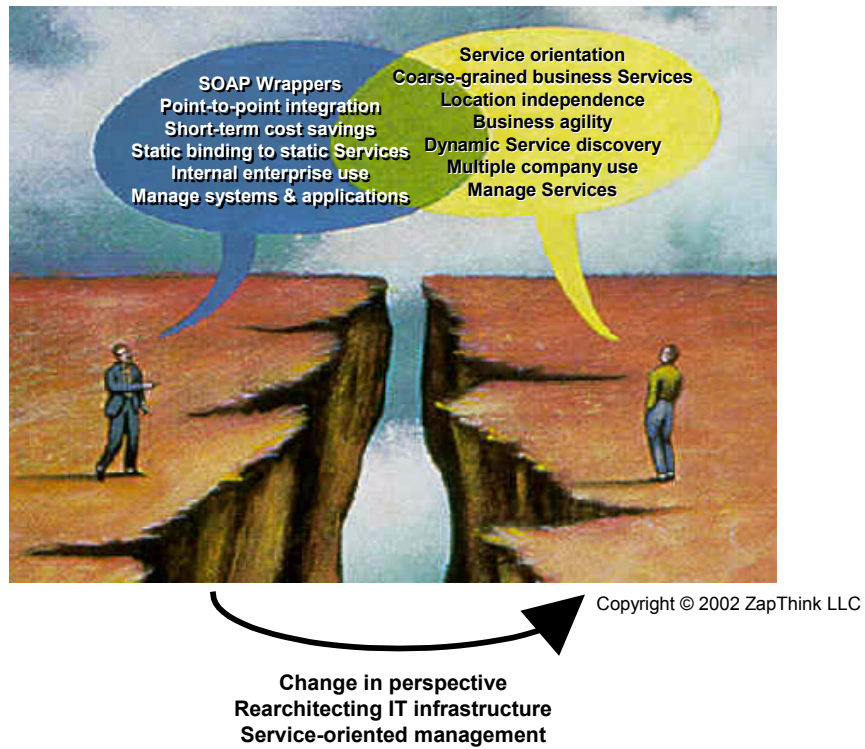
- *A successful SOA is always in flux* – To visualize how an SOA is supposed to work, it's better to think of a living organism rather than the traditional "building a house" metaphor that gave software architecture its name. The everyday normal state of affairs is an IT environment that is undergoing constant change, and as a result, the work of the IT department is never done. House-building assumes a state of completion and the ability to craft a design that remains the same over time, which is rarely the case in any business environment.

Building an SOA in the enterprise, however, is a difficult task. Companies should take a step-by-step approach to achieving Service orientation, first by creating and exposing individual Services, and then by combining them into a Web Services network, which offers the key functionality of an SOA. Only then can companies access and combine the available business Services into flexible business processes that provide agility to the business. Furthermore, an essential element of building an SOA is having software that can enable the combination of individual Services, keep them running, and enable the IT organization to manage those Services. That is the role of a Web Services management solution.

Crossing the chasm from individual Web Services

The obvious question, then, is how an enterprise should move from its early or pilot Web Services projects to a strategic architectural approach that will provide the agility it desires. Fundamentally, the answer begins with a change in perspective, as illustrated in the following figure:

Figure 1: The Chasm to Service Orientation



On the left of Figure 1 is the point-to-point Web Services perspective, which centers on providing Web Services interfaces to software components in order to solve point-to-point integration problems. On the right of Figure 1 is the Service orientation perspective of Web Services. In this perspective, IT provides coarse-grained business Services in a location and connection-independent way, in other words, in a *Web Services network*. The nuts and bolts of the software that makes such Services available, including Web Services management software, takes place behind the scenes from the business user, because the applications and systems that actually provide the Service functionality are fully encapsulated and separated from the Web Service consumers.

Web Service networks: putting the pieces in place in the enterprise

A Web Services network, therefore, hides the technical underpinnings of the enterprise’s business Services. For enterprises to offer such Services, they must build loosely coupled, Service-oriented architectures that both *encapsulate* and *virtualize* the underlying applications and systems. The concepts of encapsulation and virtualization, therefore, are fundamental to building SOAs.

Encapsulation is one of the fundamental principles of computer programming, and it is just as important to Web Services. A software object is encapsulated when its inner workings are hidden from the outside world. All interactions with such an object take place through its interface via public method calls. Encapsulation is important because it breaks up large software projects into bite-sized chunks. Different objects can be assigned to different developers, and as long as they agree on those interfaces, they can all work in parallel. Furthermore,

Enterprises must build loosely coupled, Service-oriented architectures that both encapsulate and virtualize the underlying applications and systems.

By encapsulating software components, applications, and underlying systems with Web Services interfaces and then composing these fine-grained functional Web Services into coarse-grained business Services and putting them on a Web Services network, companies will have IT infrastructures that provide business functionality that meets the needs of business better than before.

encapsulated objects are defined by their functionality. As long as they work as they should, nobody has to know what's going on inside.

However, what happens when a developer has to change the interface of an object? In this case, all the other developers have to know about the modification to the interface and potentially change their own applications in response. This is precisely the problem of tight coupling. If those objects expose their functionality via Web Services interfaces, however, then the developers don't have to know ahead of time the specifics and changes to interfaces. Rather, they just have to agree to use standards like SOAP and WSDL. Now their objects can work with each other, but much more importantly, can work with other objects that the developers have no knowledge of, as long as those new objects also expose their functionality via Web Services interfaces. That's the power of loose coupling.

When individual Web Services are combined into coarse-grained business Services, this virtualization process is known as *Web Services composition*—essentially, the business Services offer the functionality of several individual Services, as well as the functionality of other software. The composition of Web Services, therefore, heralds the next evolutionary step in this inexorable progression to the next level of abstraction—Service orientation. By encapsulating software components, applications, and underlying systems with Web Services interfaces and then composing these fine-grained functional Web Services into coarse-grained business Services and putting them on a Web Services network, companies will have IT infrastructures that provide business functionality that meets the needs of business better than before. Such Web Services networks are necessary for enabling the loose coupling that is essential for providing business agility. However, at its core, the Service-oriented view of information technology is just a layer of abstraction—a virtualization of underlying software components and applications. Every Web Service is still software—still nothing but ones and zeroes—at its core.

Service orchestration: providing value to business users

Once the enterprise has composed its individual Web Services and other software functionality into coarse-grained business Services and placed them on a managed Web Services network, how does the business make use of those Services? Line of business users are interested in creating, executing, modifying, and managing business processes, which are essentially sequences of actions or events that can include interactions with the Web Services network. Such activities are collectively called *Web Service orchestration*. Because the business Services on the network are built following the principles of SOAs, they are inherently flexible. As a result, the business processes that incorporate those Services are flexible as well.

This balance between the technology on the one hand and the business on the other enables the business to drive the requirements for the technology in an environment of flux. Traditional approaches to software architecture presuppose a traditional software development lifecycle, where users define their needs, and then IT builds and deploys the required system. In reality, this traditional, “waterfall” approach typically does not solve many of business' challenges with IT requirements, for a variety of reasons that boil down to risks that develop as a result of unknown or changing circumstances. As a result, companies react to the risks of the traditional approach by constraining the expectations of the business, essentially allowing technological risks and limitations to drive the business. Service orientation reverses this predicament, providing sufficient flexibility to allow business to drive the technology.

Business process intelligence (BPI): closing the agility loop

To make the interaction between technology and business flexible enough to enable business agility, there is one additional element needed: a feedback loop from the Web Services network to the line of business that provides visibility to the business user. It is not sufficient for the business to simply tell IT what it needs; it must also take an active role in managing the operation of the Web Services network. Furthermore, the Web Services network must provide business users with the intelligence they need to manage the business processes they control. With flexibility comes the need for vigilance, and thus a Web Services management platform must provide the enterprise with the tools it needs to maintain the smooth-running operation of its Web Services network.

Service orientation: the next major enterprise computing movement

The importance of SOAs to the development of enterprise computing is as dramatic as the business value such architectures can provide. Service orientation is the fourth major distributed computing approach to affect IT since the mid-twentieth century, after mainframe timesharing, client/server, and n-Tier/Web architectures. Each of these approaches to distributed computing does not replace the one that came before, but merely augments it. Mainframes and client/server applications still exist today. Likewise, Service-oriented applications aren't going to make Java obsolete. Instead, as technologies mature, it becomes economically practical to address a new level of business motivation.

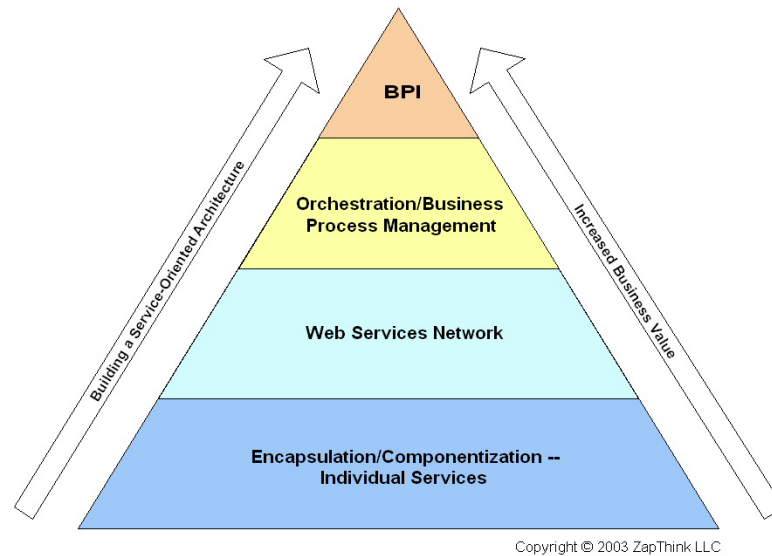
Client/server approaches only became a reality when businesses could afford PCs. The Internet's rapid rise depended on the service provider infrastructure and the availability of TCP/IP on the desktop. And now, enterprises can have the agility necessary to let business needs drive the technology. The most important conclusion to be drawn from this progression, however, is that the status of Service orientation as a distributed computing approach is every bit as important as the others that came before—but is still an evolutionary step in an ongoing process.

III. Digital Evolution: Comprehensive Web Services Management

Digital Evolution offers a practical solution to the problem of obtaining business value from Web Services. To enable a Service-oriented architecture to provide the business agility that enterprises require, it is essential to have a comprehensive Web Services management solution that can support and coordinate individual Web Services, Web Services networks, and orchestrations of Web Services, as illustrated in Figure 2 below. In addition, the management solution must provide the visibility to the business they need to manage the Web Services network and the intelligence necessary to run the business. One such solution is the *Digital Evolution Management Server*.

To enable an SOA to provide the business agility that enterprises require, it is essential to have a comprehensive Web Services management solution that can support and coordinate individual Web Services, Web Services networks, and orchestrations of Web Services.

Figure 2: Building an SOA



Single Service management

The Digital Evolution (DE) Management Server offers many of the features companies need to implement point-to-point Web Services. The DE solution manages the security of individual Services as well as monitoring and controlling the Service-level agreements that govern the requirements for the operation of the Services. The DE solution also enables load balancing and failover of Web Services, enabling companies to support Services with high availability. As companies roll out new or modified Web Services, the DE solution helps them manage multiple versions of Services. The DE solution supports both J2EE and .NET, but doesn't require an application server to operate, providing companies with a high degree of flexibility.

The level of management the DE Management Server offers is essential for running enterprise-class Web Services, even when they are used to solve point-to-point integration problems, but especially when they are used to create SOAs. As enterprises use Web Services for mission critical tasks, a Web Services management solution like Digital Evolution's becomes a critical necessity.

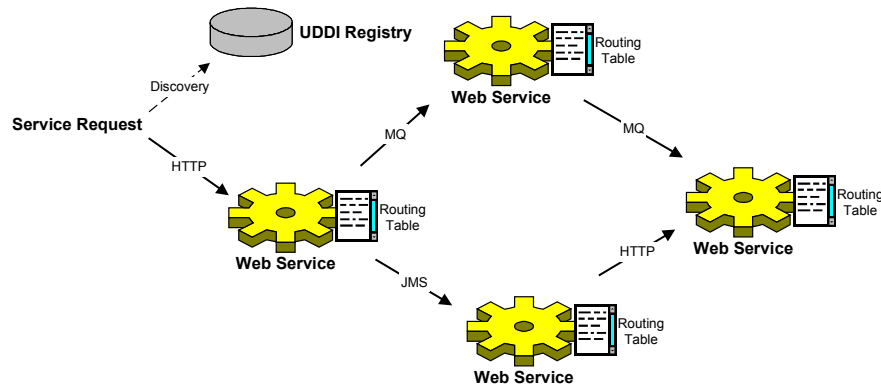
Managing Service networks

As companies look to cross the chasm to Service orientation and implement a Web Services network, they have a different set of Web Services management requirements. The DE Management server offers broad, enterprise-level security and policy management that governs the operation of all the Web Services in the network. The DE solution also supports multiple transports and routing approaches, providing the essential encapsulation of underlying transport protocols, including message-oriented middleware, HTTP, and other protocols.

The DE solution also offers a sophisticated content-based routing approach to sending messages among Web Services as shown in Figure 3 below, which is an essential part of the Web Service composition process. The routing rules can specify the most reliable, fastest, or shortest path, and can direct messages among Web services to be either synchronous (where the sender waits for a response) or asynchronous (where the sender continues to operate regardless of

when the response comes). This approach to routing enables the DE solution to offer load balancing and high availability of the individual Services in the network in a loosely coupled fashion.

Figure 3: Content-based routing in a Web Services network



The DE solution mitigates the risk facing both Web Service producers and consumers by providing security, management, versioning, interoperability, transport independence, routing, and support for multiple standards. The DE solution also incorporates a UDDI-compliant registry that supports the dynamic discovery of individual Services and provides location independence for those Services: the Web Service consumer need not know where the Service is located on the network. Finally, the DE solution offers visibility to the manager through an alert engine, as well as SNMP support for communication with existing system management products.

Orchestrating Web Services

Traditional approaches to Business Process Management (BPM) have several weaknesses that SOAs address. With conventional BPM, technical implementation teams are frequently the only personnel that have visibility into the operation of the business processes, while business managers are left out of the loop. Furthermore, conventional BPM tools tend to be overly complex, leading to excessive Total Cost of Ownership (TCO) and long implementation cycles. Most significantly, however, these tools are inflexible, reducing their effectiveness. In many cases, BPM tools become entirely useless, as the models of the processes diverge from the processes themselves.

Enterprises can overcome the weaknesses of conventional BPM by implementing a process modeling tool, such as the one offered by Digital Evolution, in conjunction with implementing an SOA. The DE solution enables active management at the business level rather than solely at the systems level. The DE solution offers both passive management of business processes (e.g., logical Service monitoring, business-level agreements, and process-level SLAs) as well as active orchestration of the Services that make up those processes. Active orchestration includes transaction management and interaction with systems that support the new Business Process Execution Language for Web Services (BPEL4WS) specification.

DE's graphical BPM tool allows users to model existing processes that span one or more systems, enabling business users to create abstract processes for

With conventional BPM, technical implementation teams are frequently the only personnel that have visibility into the operation of the business processes, while business managers are left out of the loop.

accessing data from different sources. And while active orchestration provides the most value to the enterprise because it leverages the agility advantages of the SOA, the DE solution offers monitoring and reporting separate from active orchestration for situations where such orchestration is not practical.

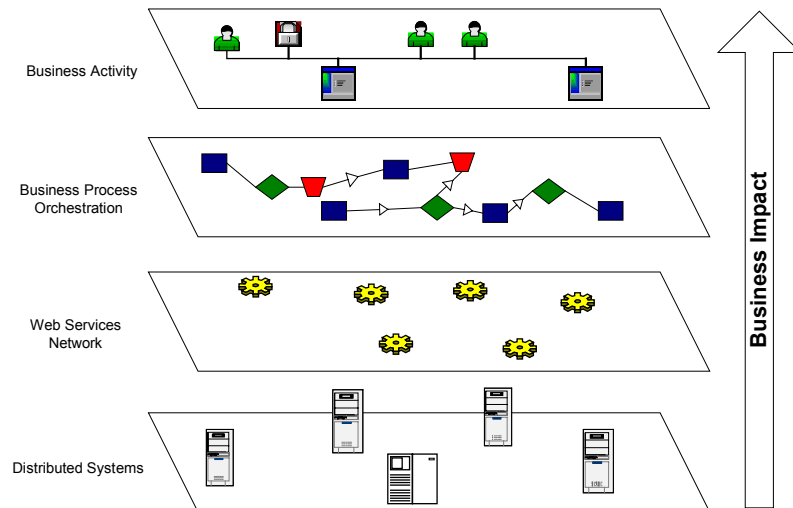
Providing business intelligence

The DE Management Server closes the business agility feedback loop by providing Business Activity Management (BAM) capabilities to the business manager, including business analytics and associated reports. This capability provides an understanding not just what has failed but what the business impact of that failure will be, which can affect the prioritization of certain activities, particularly in complex environments. Real-time business intelligence is a by-product of this capability.

IV. Conclusion: the Need for Comprehensive Web Services Management

Service-oriented architectures can provide an agile IT environment that can enable business agility in the enterprise. However, building, running, and managing such architectures is often a complex task. Enterprises can successfully build SOAs by following a series of incremental steps, each of which provides increased business value, as shown in Figure 4 below.

Figure 4: Building a Service-oriented architecture



First, enterprises should apply Web Services to solve point-to-point integration problems. This first step will both reduce the cost of integration and build Web Services expertise within the enterprise. Next, companies must shift their perspective on their IT infrastructure to provide the foundation necessary for building a SOA based on a Web Services network. Once the Web Services network is operational, companies can provide business users with the tools they need to manage business processes based on orchestration of Web Services. Finally, companies need a feedback loop between the technology and the business that enables business management to monitor and control the business activity within the enterprise.

Enterprises should look for Web Service management solutions that are comprehensive enough to support the entire path toward the fully operational SOA.

Web Services management is essential for making this entire approach successful. Enterprises should look for Web Service management solutions that are comprehensive enough to support the entire path toward the fully operational SOA. Companies must take a step-by-step approach to building an SOA, and it is critical that this initiative show solid business value at each step. It is essential, therefore, that enterprises leverage the power of a comprehensive Web Service management solution that supports the elements of the SOA as companies make the transition to Service orientation. Digital Evolution offers such a solution. Furthermore, Digital Evolution is one of the only vendors to provide an integrated product line that offers management, security, orchestration, business process management, and business process intelligence. Their comprehensive approach simplifies customers' vendor relationships, streamlines integration, and lowers their Total Cost of Ownership.

Copyright, Trademark Notice, and Statement of Opinion

All Contents Copyright © 2003 ZapThink, LLC. All rights reserved. The information contained herein has been obtained from sources believed to be reliable. ZapThink disclaims all warranties as to the accuracy, completeness or adequacy of such information. ZapThink shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The reader assumes sole responsibility for the selection of these materials to achieve its intended results. The opinions expressed herein are subject to change without notice. All trademarks, service marks, and trade names are trademarked by their respective owners and ZapThink makes no claims to these names.

About ZapThink, LLC

ZapThink is an IT market intelligence firm that provides trusted advice and critical insight into XML, Web Services, and Service Orientation. We provide our target audience of IT vendors, service providers and end-users a clear roadmap for standards-based, loosely coupled distributed computing – a vision of IT meeting the needs of the agile business.

ZapThink's role is to help companies understand these IT products and services in the context of SOAs and the vision of Service Orientation. ZapThink provides market intelligence to IT vendors who offer XML and Web Services-based products to help them understand their competitive landscape and how to communicate their value proposition to their customers within the context of Service Orientation, and lay out their product roadmaps for the coming wave of Service Orientation. ZapThink also provides implementation intelligence to IT users who are seeking guidance and clarity into how to assemble the available products and services into a coherent roadmap to Service Orientation. Finally, ZapThink provides demand intelligence to IT vendors and service providers who must understand the needs of IT users as they follow the roadmap to Service Orientation.

ZapThink's senior analysts are widely regarded as the "go to analysts" for XML, Web Services, and SOAs by vendors, end-users, and the press. They are in great demand as speakers, and have presented at conferences and industry events around the world. They are among the most quoted industry analysts in the IT industry.

ZapThink was founded in October 2000 and is headquartered in Waltham, Massachusetts. Its customers include Global 1000 firms, public sector organizations around the world, and many emerging businesses. ZapThink Analysts have years of experience in IT as well as research and analysis. Its analysts have previously been with such firms as IDC and ChannelWave, and have sat on the working group committees for standards bodies such as RosettaNet, UDDI, CPExchange, ebXML, EIDX, and CompTIA.

Call, email, or visit the ZapThink Web site to learn more about how ZapThink can help you to better understand how XML and Web Services impact your business or organization.

ZAPTHINK CONTACT:

ZapThink, LLC
11 Willow Street, Suite 200
Waltham, MA 02453
Phone: +1 (781) 207 0203
Fax: +1 (786) 524 3186
info@zapthink.com

