

# zapthink white paper

## WEB SERVICES MANAGEMENT

*SUCCESSFULLY ARCHITECTING THE FUTURE OF  
YOUR BUSINESS*





## WEB SERVICES MANAGEMENT

### *SUCCESSFULLY ARCHITECTING THE FUTURE OF YOUR BUSINESS*

November 2002

Analyst: Jason Bloomberg

#### Abstract

Today, companies are reducing the cost of integration by using Web Services to solve point-to-point integration problems inside the enterprise. However, the most important business value of Web Services lies not with the creation and deployment of individual Services themselves, but the new approaches to architecting IT infrastructures that they herald. These new architectures are loosely coupled, standards-based and Service-oriented. In such architectures, software functionality is exposed as business-oriented Services in a way that decouples the Service from the underlying software. The systems and applications that provide the Service are transparent to the systems that consume the Service.

Companies that adopt such architectures will have information technology infrastructures that provide the flexibility necessary to enable them to leverage changing business environments to their best advantage. To move from simple, point-to-point integration to Service-oriented architectures, however, calls for a new approach to managing the software within an organization, known as *Web Services Management*. With Web Services Management, companies can build and support enterprise-class Service-oriented architectures that enable true business agility.

All Contents Copyright © 2002 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.	Introduction.....	4
	Today's Key Business Driver: Business Agility.....	4
	Key Technology Driver: Efficient Integration .....	5
	Service-Oriented Architectures: The Key to Business Agility .....	5
II.	Web Services Adoption.....	6
	The Web Services Adoption Curve.....	6
III.	Web Services Management: The Key to the SOA .....	7
	The Five Categories of Web Services Management Functionality .....	7
	The Web Services Management Catch-22: Where to Start? .....	9
IV.	Actional SOAPswitch and SOAPstation: Two Layers of an Effective Web Services Architecture.	10
V.	Next Steps for the Enterprise.....	12
	The Enterprise Imperative .....	12

## I. Introduction

In today's difficult IT environment, it is more important than ever to understand the business benefits a company can expect from any technology investment. There is so much hype surrounding Web Services that it is essential for companies looking at Web Services to separate the reality from the hype. Instead of looking at Web Services as a new technology and asking what the applications for the technology are, smart IT executives should understand the fundamental business drivers facing their organizations and then look to identify how particular technologies can address those drivers cost effectively.

What, then, are the key business drivers facing enterprises today that Web Services might address? The most pressing and critical business driver today is the need for *business agility*. For too long the limitations of inflexible technology have restricted how companies can deal with changing business environments. If Web Services technologies can provide that long sought-after IT flexibility, then business imperatives will finally be able to drive technology decisions, instead of the other way around.

*The most pressing and critical business driver today is the need for business agility.*

### Today's Key Business Driver: Business Agility

*Business agility* is the ability for businesses to proactively manage change and use changing business environments to their advantage. Change comes in many forms: changes in the marketplace, in technology, and in the world at large. Companies that can make effective use of a changing environment are better able to compete and thrive, especially in tough markets.

Technology is the most important factor in many companies' ability to be agile, for two reasons. First, technology has become so intertwined with the business processes at most companies that changing a process often begins with changing the technology. Second, because technology is often expensive, difficult, and complex, technical issues often become the bottleneck that prevents companies from becoming agile. And of all the technology issues facing companies today, the one that is most likely to be a bottleneck is *integration*. The ostensibly simple act of getting two or more different systems to talk to each other—in a flexible, agile way—has been a surprisingly persistent problem since the advent of distributed computing decades ago. Simplifying integration so that it is more cost effective and flexible will go a long way to remove the technology bottleneck. Clearly, efficient integration is the key technology driver for enterprises today.

*Of all the technology issues facing companies today, the one that is most likely to be a bottleneck is integration.*

### 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](http://www.zapthink.com/credit) and enter the code ACTWSM. 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](mailto:info@zapthink.com).

What companies need is a loosely coupled approach to integration—one that does not require control of systems on both ends.

Companies that successfully rearchitect their IT infrastructures to take advantage of Service-oriented architectures are able to remove the integration bottleneck, and connect different systems in a flexible, cost-effective manner.

### Key Technology Driver: Efficient Integration

If information technology is to deliver business agility, then integration among systems must be easy to configure, easy to change, and cost effective. Why, then, are the problems with integration still plaguing companies, even though distributed computing has been around for a generation or more? At one level, the cause of these integration problems is the lack of standard ways to program different systems to communicate. Given two different systems, the traditional integration approach is to write programming code for each system that is specifically written 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.

What companies need is a *loosely coupled* approach to integration—one that does not require control of systems on both ends. However, for such an approach to be a reality, there must be standard, established ways of handling integration so that any company that follows the standards can be confident that their systems will interoperate with other targeted systems. This standards-based approach to loosely coupled integration is what Web Services are all about.

### Service-Oriented Architectures: The Key to Business Agility

Loosely coupled, standards-based Service-oriented architectures (SOAs) are an approach to distributed computing that regards software resources as Services available on the network. Consumers of these Services (which are the systems that wish to integrate with the Services) can find and connect to the desired Services in a loosely coupled fashion. Ideally, the Services available on the network expose the functionality of the underlying software in terms of business concepts. Web Services that offer business-oriented functionality instead of detailed technical functionality are known as *coarse grained*. Such coarse granularity is one of the most important features of SOAs.

In general, SOAs offer the following five 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 underlying systems are encapsulated, so that their inner workings are hidden from the Service consumers.
- Authentication and authorization of Service consumers, and in general all security functionality that applies to the Services, can also be provided via Web Services interfaces.
- Web Services consumers can find and connect to available Services dynamically.

Companies that successfully rearchitect their IT infrastructures to take advantage of Service-oriented architectures are able to remove the integration bottleneck, and connect different systems in a flexible, cost-effective manner. With the

integration bottleneck removed, then, such companies are able to achieve their desired business agility. However, such rearchitecting is easier said than done.

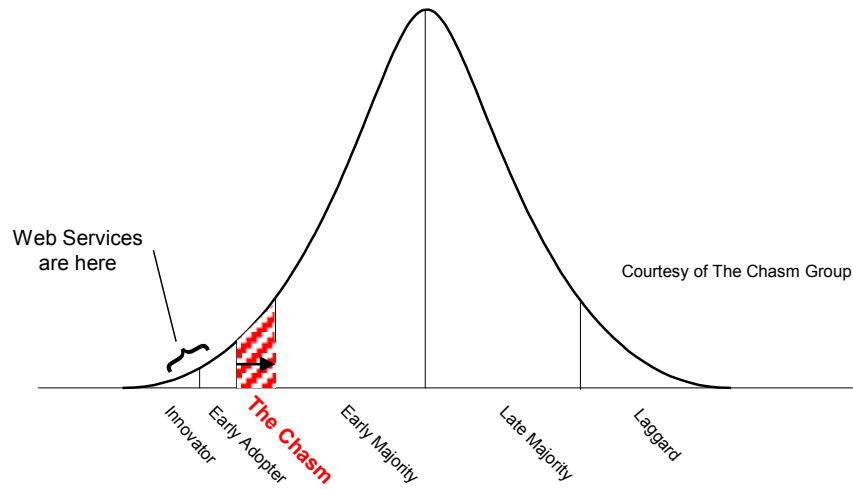
## II. Web Services Adoption

While there is the promise of substantial cost savings resulting from more efficient integration, much work remains to tackle the difficult rearchitecting work involved in moving toward a Service-oriented architecture. The most important change is the one of *perspective*: changing how companies think about IT from a point-to-point integration model to a Service-oriented model. Once companies make this conceptual change, then the necessary technical changes start to fall into place. But make no mistake: rearchitecting at the enterprise level is a difficult undertaking.

*Make no mistake: rearchitecting at the enterprise level is a difficult undertaking.*

### The Web Services Adoption Curve

This rearchitecture process is the key to the widespread acceptance of Web Services in the enterprise, as shown in the following figure.



**Figure 1: The Web Services adoption curve**

Figure 1 shows the familiar technology adoption curve, illustrating how only a few companies are innovators or early adopters of a new technology. The new technology must then cross a “chasm” to get to the majority phases, where most companies come to accept the new technology. Today, Web Services is in the innovator and early adopter phases. To cross the chasm to widespread adoption of Web Services, companies must move toward adopting SOAs.

*To cross the chasm to widespread adoption of Web Services, companies must move toward adopting Service-oriented architectures.*

Today, most companies use Web Services to solve straightforward integration problems. Often such undertakings are pilot projects, or have limited scope and importance within the enterprise. However, the number of such Web Services projects is increasing quite rapidly. For a company to take a number of scattered Web Services and coordinate them into coarse-grained business Services that have enterprise-level quality, they must utilize an underlying Web Services infrastructure that can handle a variety of technical issues behind the scenes. Software products that provide such an infrastructure fall into a category of software known as *Web Services Management* solutions.

Web Services Management solutions bridge the gap between the underlying systems and the Services that run on top of them.

### III. Web Services Management: The Key to the SOA

In today's IT environment, IT management is tightly coupled to the systems being managed. System management products provide visibility and control into the various systems that make up an enterprise's IT infrastructure. In an SOA, however, what are most important are the *Services*, not the systems *per se*. It is still just as important to manage the systems that underlie the Services, but even more important is the ability to manage how the systems enable business Services to function as they should—in a location independent, coarse-grained fashion. Web Services Management solutions bridge the gap between the underlying systems and the Services that run on top of them.

#### The Five Categories of Web Services Management Functionality

Bridging the gap between systems and Services is logically more complex than simply managing the underlying systems. As a result, several software vendors are offering new products, called *Web Services Management Platforms* that provide a range of functionality for managing SOAs and the Web Services they contain. These features fall into five general categories: system management, lifecycle management, business management, security management and SOA enablement, as shown in the figure below.

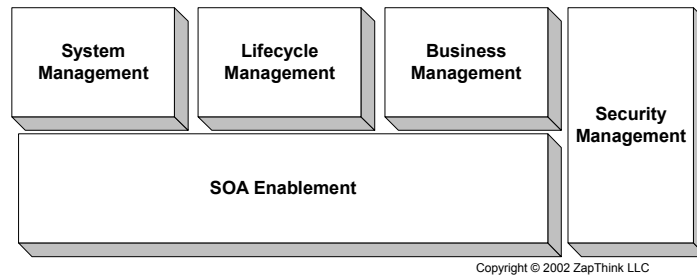


Figure 2: The five categories of Web Services management functionality

The first category of management that Web Services management must address is system management. Since Web Services run on software that in turn runs on systems, all Web Services management platforms must begin with system management. The functions that fall into the system management category include:

- *Monitoring* – insuring that underlying systems are up and running.
- *Alerts* – notifying the appropriate people or other systems when there's a problem with a system.
- *Auditing and reporting* – tracking the usage of Web Services and making that information available to reporting systems.
- *Service-level agreement (SLA) and quality of service (QoS) management* – tracking the quality of service offered to particular customers (uptime, latency, etc.) and adjusting various performance criteria to insure that each customer is getting the contracted service level.
- *Exception management/problem resolution* – handling problems that occur by following policies set out for dealing with such problems.

- *Root cause analysis* – digging down through various layers of system functionality (Web Service, application, application server, network, operating system, hardware) to find the actual cause of particular problems.

In addition to making sure the underlying systems are working properly, Web Services management platforms often handle the process for putting new Web Services into production, and dealing with them when they change. The functions fall into the lifecycle management category and include:

- *Provisioning* – putting new Web Services into production and helping Web Service consumers access them properly.
- *Versioning* – managing different versions of Web Services that may be in production at the same time.
- *Deprecation* – retiring out-of-date Web Services gracefully, without breaking any Web Service consumers that may be accessing them.
- *Dependencies* – handling changing Web Services when one Web Service may depend on another.
- *Configuration management* – managing different configurations of one or more Web Services as companies upgrade or change the Services.
- *Web Service emulation* – simulating sets of Web Services for the purposes of testing.

IT administrators are typically in charge of the system and lifecycle management functionality of a Web Service management platform. However, several platforms also provide visibility and control for the line of business manager. These managers are concerned with the following functions, which fall into the business management category:

- *Business activity monitoring and decision support* – accessing the critical business information in an IT infrastructure needed to make business decisions.
- *Message prioritization/differentiated QoS* – controlling which customers get access to particular Web Services, especially when those Services are not able to serve all customers equally.
- *Billing and metering* – determining the use of particular Web Services and feeding that information to a metering, usage control, or billing systems.

Overlapping the area of Web Services management is the Web Services security segment of the market. In many ways, security is a category in its own right (which is why they are off to the side in Figure 3), but many Web Services management products also manage Web Services security. The sorts of functions that such products manage include:

- *Authentication and access control* – making sure that only authorized Web Services consumers can access individual Services.
- *Confidentiality* – encrypting messages to make them more secure. May also involve decrypting incoming messages to inspect them and then reencrypting them.



Platforms that provide SOA enablement capabilities help companies take collections of fine-grained Web Services and roll them up into coarse-grained Business services.

- *Malicious attack protection* – preventing unauthorized access to Web Services, including those that are actively seeking to penetrate the Services.
- *Non-repudiation* – providing an audit trail for incoming or outgoing messages to guarantee their delivery.

Putting all the system, lifecycle, business and security management functionality together offers a broad range of management capabilities to the enterprise, but still does not provide the critical connection between coarse-grained business services and the systems and applications that support them. Web Services management platforms must also provide a set of functions we call *SOA enablement*. Platforms that provide SOA enablement capabilities help companies take collections of fine-grained Web Services and roll them up into coarse-grained Business services. The functions that fall into the SOA enablement category include:

- *Synchronous/asynchronous conversion* – if an underlying system is unable to respond to a Web Services request in real time, the management platform can respond with an automatic “your request is being processed” message, and then send the desired result when it is ready.
- *Encapsulation and virtualization* – taking fine-grained functional Web Services (typically wrappers for existing applications) and combining them into coarse-grained business Services.
- *Web Service instance management/resource mobility* – preserving the location independence properties of an SOA by enabling different systems in different locations to provide particular Web Services.
- *Dynamic routing* – sending Web Service consumers’ requests to different systems depending on availability, preserving location independence and improving overall system performance.
- *Protocol translation* – translation among different protocol and messaging formats including SOAP, Java messaging, message-oriented middleware like IBM WebSphere MQ (MQSeries), DCOM, and others so that systems that are not exposed as Web Services can participate fully in the SOA.

### The Web Services Management Catch-22: Where to Start?

The last of these functions—protocol translation—is particularly important for companies considering how to move to an SOA. The situation many companies find themselves in is a “Catch-22”: why buy Web Services management before there are many Web Services in production, and how can you run many Web Services without being able to manage them? The solution to this Catch-22 is to manage existing systems and services as Web Services gradually roll out.

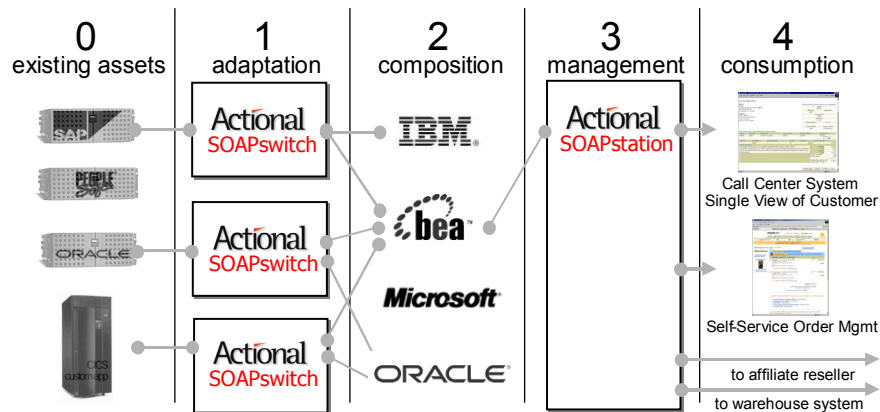
For protocol translation approach to work, it is essential for the Web Services management platform to leverage protocol translation in a robust, loosely coupled manner—in what is often called an *adaptation layer*. Companies that may only have a few trial Web Services implementations in their IT environments are then able to use this adaptation layer to implement the Web Services management platform, obtaining immediate value by opening their existing applications as Web Services. Then, as they tackle the more difficult task of rearchitecting their systems to create an SOA, they will have the management infrastructure in place to support it.

It is essential for the Web Services management platform to leverage protocol translation in a robust, loosely coupled manner—in what is often called an adaptation layer.

In fact, Web Service-enabling existing systems provides a key value proposition for Web Services by getting increased value out of the data and functionality locked in legacy applications. Today's legacy IT projects typically target a very specific bit of information on the legacy system, for example, inventory or account balance. By providing a Web Services interface to the legacy system, far more of the data stored within that system is potentially available for access via a Web Services interface. To make true use out of these newly unlocked data assets, however, companies must have the Web Services management layer on top of the adaptation layer to provide the infrastructure necessary to use such Web Services on an enterprise level.

#### IV. Actional SOAPswitch and SOAPstation: Two Layers of an Effective Web Services Architecture

Actional is unique among Web Services vendors in that they have a product called SOAPswitch that provides the adaptation layer that is critical for providing protocol translation. SOAPswitch provides a loosely coupled interface between existing assets, including databases, legacy applications, and other enterprise applications on the one hand, and the application server platform tier (which they call the composition tier) on the other, as shown in the figure below.



**Figure 3: The Actional Web Services Management Architecture**

Another product, SOAPstation, then provides the Web Services management layer between the composition tier and Web Service consumers. SOAPstation provides Web Services management features in each of the functionality categories as explained above. The combination of SOAPswitch and SOAPstation completes a fully functional Web Services architecture, filling in the gaps left by current platform vendors. The combination of SOAPswitch and SOAPstation provide:

- *Lifecycle management: service provisioning, versioning, deprecation, and configuration management* – SOAPstation provides service provisioning and associated change management features, including the creation of service access points that have varying protocols and security models, and the versioning and deprecation of deployed services for both synchronous (client/server) and asynchronous (publish/subscribe) Web Services. When evaluated against the full set of lifecycle management features outlined in section 3 of this paper,

SOAPstation provides a critical mass of features to manage Web Service life cycles cost effectively.

- *System management: monitoring, alerting, auditing, and reporting, exception management and root cause analysis* – Both SOAPstation and SOAPswitch provide robust activity monitoring and reporting, with a user interface for administrators. They monitor system and service performance, alert administrators when QoS levels are being encroached upon, and provide drill-down from auditing to event logs for root-cause analysis. In addition, they provide end-to-end Quality of Service by delivering features such as load-balance and back-up routing, proactive service availability monitoring and reliable delivery of messages to both Web Service and non-Web Service enabled systems. When deployed together or individually, SOAPstation and SOAPswitch offer end-to-end visibility and management of complex Web service deployments.
- *Business management: business activity monitoring, message prioritization and billing and metering* – Both SOAPstation and SOAPswitch provide configurable monitoring, statistic gathering, and reporting to support business users information needs. In addition, all of this information can be logged to external databases and integrated with third-party reporting, billing and analysis tools.
- *Security management: security framework, authentication and authorization, rules-based access control and non-repudiation* – Actional SOAPstation offers a flexible security framework for managing both credential-based and business-based access on a per-Service basis. SOAPstation offers XML Signature support, multiple approaches to sender authentication and authorization, and rules-based access control that integrates with a variety of existing security infrastructure approaches based on LDAP and third-party single sign-on products. In addition, SOAPstation allows for integration with non-standards-based security systems. Actional SOAPswitch allows legacy systems to become securely enabled for Web service access. Actional's solutions allow companies to securely deploy Web services in today's heterogeneous and evolving security infrastructures.
- *SOA enablement: dynamic routing, encapsulation and virtualization and protocol transformation* – Actional's two-part approach to Web Services management provides an additional layer of coarse granularity that other Web Services management platforms lack. SOAPswitch provides remote creation and configuration of Web Services via a GUI, and manages the publishing of Services to UDDI-compliant servers. It also supports different failover techniques that add a measure of location independence. SOAPstation also offers configurable, rules-based routing as well as message and protocol transformations, including support for asynchronous Web Services.

In general, ZapThink believes that Actional's two-pronged approach to Web Services management offers a complete solution for companies who are moving toward enterprise class Web Services and SOAs.

*In the short term, IT departments must reduce costs, and Web Services offer a straightforward approach to reducing the cost of integration. In the long term, IT must empower the enterprise to take advantage of changes in the business environment.*

## V. Next Steps for the Enterprise

In the reality of today's IT world of tight budgets and laser focus on the bottom line, what circumstances would lead an enterprise to believe a Web Service Management platform like Actional's deserves serious consideration? The answer is both tactical and strategic. In the short term, IT departments must reduce costs, and Web Services offer a straightforward approach to reducing the cost of integration—one of the most expensive items on today's IT budget. In the long term, IT must increasingly be a profit center for the enterprise and empower the enterprise to take advantage of changes in the business environment. However, most IT executives today do not have the luxury of thinking long-term.

Naturally, smart executives realize that business agility and strategic planning are critical for the long-term success of their organizations, regardless of the short-term constraints today's environment provides. Such executives must find solutions to their short-term cost containment problems that also work to address their long-term concerns, as well. Web Services Management is just that kind of solution: it enables companies to take advantage of the cost savings of Web Services today, while preparing the company to more effectively leverage its IT architecture in the future.

### The Enterprise Imperative

Web Services are the next evolutionary development in the field of distributed computing, building upon the lessons learned from client/server, n-Tier architectures, and object-oriented programming. In many ways, Web Services are not new; they simply offer a new approach to solving existing problems with existing tools. As such, they offer no panacea; companies who do not understand the true value of Web Services and Service-oriented architectures risk simply repeating the mistakes of the past.

Fundamentally, there's no way to cross the Web Services chasm without Web Services management. Using Web Services to provide simple, point-to-point integration solutions will save money in the short term, but companies will only realize long-term value from Web Services if they cross the chasm to Service-oriented architectures. And for their SOAs to offer enterprise-class business functionality that enables the agile enterprise, they must have Web Services management.

Actional's products offer a flexible and powerful solution to the challenges of enabling and managing scalable Web Service deployments. Organizations who are serious about succeeding with Web Service based SOAs would be well advised to give Actional's product line meaningful consideration.



## Copyright, Trademark Notice, and Statement of Opinion

All Contents Copyright © 2002 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](mailto:info@zapthink.com)

