zapthink
white paper

IMPLEMENTING SECURE WEB SERVICES IN BUSINESS-TO-BUSINESS ENVIRONMENTS

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

IMPLEMENTING SECURE WEB SERVICES IN B2B ENVIRONMENTS

June 2004
Analyst: Ronald Schmelzer

Abstract

Businesses are increasingly seeing Web Services as both a practical and economical way to solve a number of long-standing integration issues. While such companies are using many Web Services and Service-Oriented Architecture (SOA) approaches to solve integration issues behind the corporate firewall, many companies are also finding significant success in deploying Web Services to solve their business-to-business (B2B) integration challenges.

Yet, successful deployment of B2B Web Services involves careful consideration of a number of significant issues, not the least of which is making sure that the Services interactions between companies are secured and reliable. This whitepaper illustrates how companies can surmount their security challenges as they roll out B2B Web Services solutions. Showing how different industries have approached their B2B Web Services deployments, this paper explains how companies can tackle different aspects of security, from confidentiality to authentication and identity, in a straightforward, cost-effective manner that meets their integration needs.

I. Implementing Web Services for Business-to-Business Integration

Since the dawn of IT, enterprises have faced the challenge of connecting with their partners and their disparate systems in the IT ecosystem in a manner that is cost effective, manageable, efficient, and secure. Companies face the requirement of integrating different systems within the enterprise in order to accomplish a wide range of critical business needs, such as connecting with suppliers, partners and customers, gaining a better understanding of their business operations, and making better use of existing systems by continuing to find ways to leverage these systems for new applications.

Over 70% of today's IT budgets are dedicated to solving integration issues.

Integration is also a critical challenge at the heart of most enterprises' key business applications, such as Customer Relationship Management (CRM), Supply Chain Management (SCM), e-Business and e-Commerce applications, and enterprise portals. The need for two or more of these applications to communicate forms the basis of the application integration problem. ZapThink estimates that over 70% of today's IT budgets are dedicated to solving integration issues. In some industries, such as financial services and insurance, this percentage is even higher. As such, solving integration problems represents a significant business opportunity for enterprises as well as IT software and services vendors that are seeking to address integration issues.

In the past, IT departments have integrated systems by implementing a proprietary middleware tier such as those provided by traditional Enterprise Application Integration (EAI) or Business-to-Business Integration (B2Bi) solutions. These solutions are built primarily on proprietary or system-specific messaging platforms that aim to provide a complete, end-to-end platform for integrating and communicating with various business components. The typical method for accessing these systems is through a wide assortment of pre-built adapters that provide bidirectional connectivity to many types of applications and data sources, such as enterprise software applications, databases, file systems, and directories, as well as mainframe and other legacy applications. In simple terms, the way these integration solutions work is by extracting or inserting data from these various adapter-enabled systems, transforming the data to a different format, and then shipping the transformed data to their destination.

The primary means for integrating different businesses and organizations throughout the 1980's and 1990's was (and still is) Electronic Data Interchange (EDI), an aging data format and networking specification for B2B integration. While EDI has gained widespread acceptance, it is a fairly rigid language that accepts little modification to its data format. In addition, despite how rigidly EDI defines documents, there is still a large degree of variability among document formats that leads to semantic ambiguity when implementing EDI. Users must resolve these ambiguities in a tedious, manual manner, which can be quite cumbersome to the party that must deal with dozens, hundreds, or even thousands of trading partners.

Yet, EDI gained significant traction for a wide range of industries because it not only simplified interactions with third parties, but also addressed the critical security and reliability needs of end users. Through the use of the Value-Added Network (VAN), companies could interact using EDI-formatted messages with large numbers of suppliers and partners without having to worry about handling security and reliability on a point-to-point basis with each trading partner. The VAN served as a "store-and-forward" network that provided secured connections between partners and the network, and provided reliable, guaranteed delivery for documents sent on that network.

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

These traditional EAI and B2Bi approaches, while meeting the needs of enterprises for decades, are expensive, complicated, and cumbersome ways of integrating an ever-increasing set of applications, systems, and businesses. Traditional EAI and B2Bi approaches are fundamentally *brittle*, since they implement their integration through point-to-point, *tightly coupled* mechanisms – meaning that both ends of the integration scenario must be controlled in order for the integration to happen reliably. If either party changes their implementation of any system connected in a tightly coupled integration environment, the overall integration will fail.

The Internet has no inherent security or reliability functionality to guarantee secure, reliable interactions between partners.

Finally, the emergence of the Internet promised not only to simplify dealing with multiple systems, but also to reduce the cost of connecting to various business endpoints. Yet, while providing the communication backbone for enabling B2B integration, the Internet itself is insufficient to handle the integration needs of most companies. For one thing, the Internet has no inherent security or reliability that matches the functionality of the EDI VAN to guarantee secure, reliable interactions between partners.

Furthermore, the primary challenge in working with various business and industry endpoints is that each business and industry has different data requirements, and even worse, defines the same things in different ways. These semantic issues were made significantly worse by the use of different, and sometimes arbitrary, data formats. As a result, in order to realize the promise of significantly improved economics for integration, companies are looking to adopt *loosely coupled, standards-based* integration approaches. And this is precisely what standards-based Web Services integration, leveraging emerging trends in Service-Oriented Architectures (SOAs), offers.

Web Services: Standardizing the Interfaces between Businesses

The movement to standards-based, loosely coupled integration technologies and architectures represented by XML, Web Services, and Service-Oriented Architectures heralds a broad set of changes in today's information technology environment. XML is fast becoming the *lingua franca* of disparate, heterogeneous information on the network, and Web Services, based on XML, represent a new, open standards-based approach to getting systems to integrate with each other.

The main reason why XML-based Web Services are well suited to addressing different integration challenges is that they are capable of representing data and processing information in an application-neutral, open, and extensible manner. Therefore, vendors aren't locked into proprietary choices for integration technology. Vendor and platform lock-in are the primary reasons for much of the integration nightmare that exists today. If there was a single open, extensible mechanism for data interchange, there wouldn't be as much of an integration challenge.

Web Services and SOA approaches advocate solving critical integration problems by providing standards-based, loosely coupled interfaces for accessing application functionality and data

Web Services and SOA approaches advocate solving critical integration problems by providing standards-based, loosely coupled interfaces for accessing application functionality and data. Rather than planning in advance how a specific application will tie into another application, Web service architectures promote the concept that developers can now think about how a specific application exposes itself as services to any application that cares to speak to it. Web Services allow arbitrary applications, systems, and data stores to communicate without concern as to the other system's internal implementation. Thus, it is no surprise that enterprises and IT vendors alike are latching onto Web Services as the primary means for solving many integration issues.

The business benefits of standards-based, loosely coupled integration are clear:

- Enterprises can *reduce their cost of integration* because the interfaces between their systems and businesses have been agreed upon in advance – reducing the dependency on complex, expensive, and/or custom integration approaches.
- Enterprises can *reduce the total cost of ownership of their heterogeneous systems* since standards-based, interoperable systems give businesses more choice of vendors and flexibility to solve their specific business needs.
- Enterprises can *realize a significantly expanded market opportunity* since rather than relying on partners and suppliers to implement specific, proprietary technology approaches, vendors can provide solutions that are assured to work in their customer's environments, allowing them to reach partners that may have been inaccessible in the past.
- Enterprises can *reduce their time to market* because they can increasingly depend on critical architectural and infrastructural elements to exist in their partners' IT environments and rely on the interoperability of those elements to reduce their need to develop time-intensive, expensive, and proprietary solutions.

In addition, in a Web Services context, there really is no technology difference between EAI (internal-facing) and B2Bi (external-facing) integration. Interfaces of any type and at any location can be abstracted as a Web Service, and as such accessed through standard interfaces. The result is that there will be little need in the future for artificial divisions between EAI, B2Bi, and data integration solution classes. From a market perspective, therefore, the currently separated integration solution markets will converge on a single approach to Service-Oriented Integration (SOI).

Web Services technologies are still relatively immature, and only now are implementation details of specifications and best practices becoming established

However, Web Services technologies are still relatively immature, and only now are implementation details of specifications and best practices becoming established. As a result, companies are struggling with finding the best ways to put Web Services-based integration approaches into practice. These companies are finding that Web Services, by themselves, are insufficient to implement B2B Web Services integration scenarios that are secure are reliable. As such, additional steps need to be taken to guarantee that their interactions with their customers, partners, and suppliers will adequately meet their business needs.

Challenge to Practical B2B Web Services Implementation: Security and Reliability

The increasing use of Web Services in the enterprise presents a new set of requirements and business drivers for the adoption new standards-based tools and technologies. With this burgeoning quantity of Web Services comes new risks and problems, as well. It is essential for today's enterprise to understand both the benefits that XML and Web Services can bring to the business, but also understand the changes that IT organizations must make to accommodate new approaches to computing and the risks associated with those approaches.

To understand the security issues facing organizations as they adopt Web Services, it's essential to apply the principles of application security. Application security contains five basic requirements, expressed in terms of the messages sent between parties. Such messages include any kind of communication between the sender (party who wishes to access an application) and the recipient (the application itself). The five requirements for application level security are:

- *Authentication*. The recipient of the message must be able to confirm the identity of the sender of the message.
- *Authorization*. The sender of a message must be authorized to send the message.
- *Confidentiality*. The contents of messages must not be available to unauthorized parties.
- *Data integrity*. The recipient of a message must be able to guarantee that a message hasn't been tampered with in transit.
- *Non-repudiation*. The sender and the recipient must be able to guarantee that the sender sent and the recipient received the message, including the time the message was sent and the fact the recipient received only a single copy.

For successful B2B integration, companies should especially focus on authorization, confidentiality, and non-repudiation.

For successful B2B integration, companies should especially focus on *authorization*, *confidentiality*, and *non-repudiation*. Authorization determines whether a user is allowed to perform the functions it requests or access requested data. Confidentiality means that an unauthorized person cannot view or interfere with a communication between two parties. XML-based Web Services is particularly vulnerable to security compromises due to its human-readable, easily parsed nature. As a result, any XML message, including SOAP messages, must be enhanced with security features including encryption, digital signatures, authentication mechanisms, and privacy controls.

When secure messages are sent, the recipient often requires that the sender can't repudiate the message, or claim that the message wasn't sent at particular date and time. Likewise, a sender would like to guarantee that a given message was received. The most common way to provide non-repudiation is through the use of digital signatures. With digital signature technology, senders can both provide evidence that a document is valid while simultaneously logging the message transactions into signed audit logs. Once an audit log has been signed it cannot be surreptitiously modified.

Data integrity builds upon the notion of non-repudiation in two ways. First, the data received must be the same as the data sent. In other words, data integrity systems must be able to guarantee that a message did not change in transit, either by mistake or on purpose. The second requirement for data integrity is that at any time in the future, it is possible to prove whether different copies of the same document are in fact identical.

Critical B2B Web Services Security Issue: Context and Identity

The fundamental problem with securing Web Services is one of *context*. Security context is a set of information about the user of a Service, including the rules and policies that apply to that user, as well as information about the business process or transaction the user is currently participating in. When the user is separated from the Service, that context is lost. A single Web Service might expose functionality from several systems and applications, and in some cases, those applications may change over time. Therefore, the security context must be maintained across the entire IT infrastructure to ensure the appropriate security to the network of Web Services.

Robust enterprise identity management solutions offer a compelling set of solutions for Web Services security context challenges.

Robust enterprise identity management solutions offer a compelling set of solutions for these security context challenges. An emerging class of identity management solution provides a centralized, policy-based approach to handling B2B security issues. These solutions manage heterogeneous Web Services-based B2B integration environments by applying and enforcing security policies across multiple disparate application silos.

Identity management solutions that meet the above needs provide a two-part approach to Web Services security: one that contains a centralized Policy Decision Point (PDP) as well as distributed Policy Enforcement Points (PEP). The PDP acts as the centralized policy store, while the PEPs enforce those policies at the endpoints, which can either be at the network perimeter, or at the interfaces to the various applications that the identity management system is securing.

Identity management is especially important to securing Web Services because it maintains the security context across a network of distributed systems and applications. *Identity management* is a set of processes for the creation, maintenance, and use of identities and their attributes, as well as credentials and entitlements, plus a supporting infrastructure for managing those identities, attributes, credentials, and entitlements. Identity management solutions typically offer directory services that provide an authoritative identity repository that contains people, organizational units, groups, and roles. The identity management solution then provides authentication and authorization based on the identity information stored in the directory, leveraging user attributes such as roles and groups.

II. Case Studies in B2B Web Services

While the use of Web Services for B2B integration is still in its infancy, a number of significant companies from many different vertical industries have realized significant value from implementing Web Services for their interactions with third-party organizations. In particular, the financial services, healthcare, and manufacturing market sectors are all aggressive implementers of emerging Web Services B2B integration approaches. In this section, we will explore how key companies in these sectors have successfully surmounted the challenges for implementing secure, reliable Web Services for B2B integration.

Case Study: Financial Institution and Federated Identity

Existing applications often have custom and brittle interfaces that do not scale or adapt to changing customer requirements.

A large multinational financial institution was interested in leveraging its brand, position, and credibility as a trusted intermediary in order to provide federated trust verification services for individuals and businesses. Prior to the advent of Web Services and the widespread adoption of Service-orientation, this institution had difficulty providing identity federation services for a number of reasons. First, the lack of widely accepted standard application and B2B interfaces as well as document formats to link applications from two different business entities presented a barrier to connecting the institution to its potential customers. Second, the existing interfaces of the financial institution were custom and brittle interfaces that did not scale or adapt to changing customer requirements particularly well.

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

Leveraging the approaches discussed in this whitepaper, the company solved its federated identity challenges by implementing an Authentication Web Service, in addition to use of standardized XML document formats for B2B communication, such as ACORD or FIXML. Specifically, the company implemented a multi-step authentication scheme that allowed users to use raw XML, SOAP with attachments, or a simple SOAP message. Based on the business relationship between the two business partners, the authentication credential can be in a variety of formats including:

- Basic Username/password in HTTP header sent over SSL
- WS-Security X509 Digital Certificates
- XML Digital Signature

Upon receipt of the initial request, the Authentication web service at the financial institution authenticates and authorizes the request against a centralized security policy store and issues a SAML token back to the requestor. This SAML token can then be utilized by other Web services, enabling secure single sign-on across business domains and boundaries.

Trusted, federated identity deepens existing business relationships and opens the possibilities for greater collaboration and revenue generating services.

As a result of implementing the above standards-based federated identity system, the financial services company realized the following benefits:

- Greater Customer Satisfaction through an Enhanced B2B Relationship: Trusted, federated identity deepens existing business relationships and opens the possibilities for greater collaboration and revenue generating services.
- Increased Business Opportunities: By utilizing standard XML document formats and open standards for sharing identity information (SAML, WS-Security), the financial institution can now engage and recruit new business partners much easily. And because the interfaces reduce administrative and developer overhead, the financial institution is in the enviable position of being able to lower IT costs and increase revenue simultaneously.

Rapid and continuous change is a constant for companies throughout the financial services industries. Financial Services Providers (FSPs), encompassing a wide range of businesses and industries including equities and fixed income trading, commodities and currencies, investment banking, retail and commercial banking, insurance, and various financial-related fields, are constantly exploring new channels, increased operational efficiency, and greater visibility into processes. At the same time, mergers and acquisitions, the accelerating pace of competition, e-Commerce and portal initiatives, the desire to increase efficiency and remove dependency on manual, paper-intensive processes, fraud prevention, globalization, and the move towards real-time processing of financial transactions are creating new challenges and business opportunities for FSPs.

Complex, heterogeneous integration environments and the dependency on aging, legacy systems further challenge implementations of solutions to these problems. While the specific nature of each FSP's business differs, the industry's main concerns are the same: how to improve informational representation and flow between systems, departments, and organizations in order to improve profitability and business agility.

In the Financial Services sector, Web Services adoption is well underway. A number of accelerating factors including the need to better integrate, regulatory and reporting requirements, and the desire for secure, reliable, and robust business-to-business interactions will help make Web Services pervasive in this market sector by 2005.

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

By providing a standard, abstract interface to product catalog information, the developers were freed from having to write the queries manually.

Case Study: Retail industry

A large consumer goods company with over 50,000 employees and operations in over thirty countries delivers product information electronically via a web site to its wholesalers and retail chains. The website includes e-Commerce capabilities as well as a rich, online product catalog. However, despite the rich capabilities of this e-Commerce application, changes to application functionality required extensive programming intervention, and as a result, responding to continuous business changes were increasingly more complex and costly.

The developer team sought to solve this agility problem by exposing product catalog information component as an externally available XML Web Service, which internal users as well as partners could utilize to enter product information directly through a loosely coupled, standards-based interface. By providing a standard, abstract interface to product catalog information, the developers were freed from having to write the queries manually.

However, the primary challenge in making this Web Service work across the organization and with its third-party partners was guaranteeing that only authorized users were entitled to use the application. As such, the company implemented a robust Web Services identity and security management system to provide only authorized access to critical business data.

The firm realized a number of key benefits by implementing this secured, externally-available Web Service including:

- *Faster Time to Market*: Through the use of an abstract Web Service, product information can be updated, changed, and distributed more rapidly, enabling for the most accurate information as well as accelerating the process for inputting new product information into the product catalogue. The Web Service is also easily reused across a wide variety of platforms to accelerate future application development.
- *Reduced Application Development and Maintenance Cost*: Exposing the catalogue functionality as a Web Service freed the development team to focus on more strategic value-added projects rather than manually coding product information into a database.

Case Study: Manufacturing Industry

In another case study for secure B2B Web Services, a large manufacturing vendor was seeking to simplify the process for its field service engineers to access critical service related data currently residing on its legacy mainframe systems. Since the workforce operated remotely, there were considerable security concerns about exposing critical data over common networks to trusted employees. To get around those critical issues of security and trust, the company relied on a proprietary, custom application to access this information. However, this solution was cumbersome, expensive, not particularly scalable, and proprietary.

The manufacturing company decided to turn to Web Services as a means to expose the service information as a Web Service by "wrapping" the mainframe with a small piece of software that exposes key information as a Web Service. To address security concerns, the company decided to protect that Web Service using XML digital signatures. Leveraging this Web Service, a client application that resides on each field service engineers PC accesses secured data across the Internet. The client application compiles the field service engineers' daily service information, bundles it into a SOAP message and signs the message using XML Digital Signature. At the mainframe, the Web service intercepts the

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

Companies should leverage existing centralized security policy stores to support Web Services, thereby providing consistent security policy enforcement and risk reduction across web applications and Web Services.

incoming requests, authenticates the request, by verifying that XML Digital Signature against a centralized security policy store.

The manufacturing company realized a number of significant benefits of using secured Web Services for their remote field force communication:

- Flexibility: Migrating away from the custom proprietary application and adopting standard Web Services and XML interfaces removed the developer team from supporting custom applications. This project also served as important validation for the rest of the company by demonstrating the value of a Service-Oriented Architecture (SOA) and how to utilize a mainframe even in the most advanced and agile of business applications.
- Improved Security: The manufacturing company increased its security over previous distributed approaches due to its migration from a proprietary username/password security mechanism to an open and cryptographically sound mechanism using XML Digital Signature. Furthermore, the company was able to leverage its existing centralized security policy store to support web services, thereby providing consistent security policy enforcement and risk reduction across web applications and web services.

The increasing automation of portions of the supply chain allows suppliers and consumers to gain increasing levels of awareness of the efficiencies in the supply chain process, and greater security and reliability in the interactions between partners. Products are tracked, via their stock-keeping unit (SKU), from the time they're produced at numerous suppliers to the time they arrive at end-user locations. This increase in automation allows the supply chain to become less linear in nature. With a unifying means for identifying and sorting goods, multiple suppliers, distribution centers, and retail outlets can be used to reach the customer. The use of electronic, automated B2B interactions that are audited, reliable, and secure reduces the need for paper to be the means for tracking these movements of goods and services.

Nowadays, the supply chain is a complex web of interactions. Each manufacturer of finished goods has relationships with dozens or hundreds of suppliers, each of which have relationships with dozens or hundreds of manufacturing customers. These interrelationships have enabled the use of dynamic supply agreements that allow companies to constantly be on the lookout for better relationships and deals. The increasing globalization of business has resulted in suppliers existing anywhere in the world, covering many different countries, languages, and time zones. This globalization has added challenges and pressures in the effort to optimize supply chains – and this is where the value of Web Services-based B2B integration is most keenly realized.

Successfully solving cross-organizational security challenges is one of the critical issues in making B2B integrations based on Web Services a reality

III. The Netegrity Solution

As detailed earlier in this paper, successfully solving cross-organizational security challenges is one of the critical issues in making B2B integrations based on Web Services a reality. In Web services deployments that remain inside the corporate firewall, the security challenges are much easier to implement and control since these environments only deal with a defined constituency such as employees. However, as the above use cases demonstrate, exposing Web services to external users outside the corporate firewall dramatically increase the security risk and complexities. This is especially true as these external deployments scale to support thousands of users. Companies are now faced with the challenges of managing security not just employees but users with different roles (wholesalers, consumers, partners, etc.) and being able to apply security and access policies consistently across all consumers of the Web services.

Security problems become even greater as companies need to manage security across a wide variety of platforms, business processes, and user populations.

Furthermore, as companies deploy more and more Web services, the security problems become even greater as companies need to manage security across a wide variety of platforms, business processes, and user populations. For this reason, enterprises need to consider centralized security solutions that can apply security across all web services, taking security enforcement out of the developers and utilizing it at runtime instead.

One company providing such security and reliability solutions that secure B2B Web Services interactions is Netegrity. Netegrity® TransactionMinder® is an identity based Web Services security application that that centralizes the authentication, authorization, and audit activities for all Web Services transactions by leveraging a set of policy-based shared Services. TransactionMinder leverages SiteMinder, Netegrity's flagship single sign-on (SSO) product, to provide a scalable solution that relies on the Netegrity Policy Server, which provides a shared, centralized point of control for Web Services identity management.

Netegrity TransactionMinder delivers security policy as a shared Service, offering centralized authentication, authorization, audit, and federation for a company's Web Services.

Netegrity designed TransactionMinder to address many of the security issues present in the B2B integration scenarios mentioned in the industry discussions above. Leveraging the Netegrity Policy Server, TransactionMinder delivers security policy as a shared Service, offering centralized authentication, authorization, audit, and federation services for a company's Web Services. TransactionMinder bases its authentication approach on message content (for example, whether a message contains a purchase order, or an electronic patient record, or a stock trade) as well as the WS-Security and SAML standards. TransactionMinder externalizes security logic outside of applications, relieving developers of the responsibility for coding security directly into their applications. The product provides a single point of access control and administration for the whole enterprise by binding XML flows to user identities. TransactionMinder supports fine-grained, document-based credential checking and generation, and it checks inbound messages for authentication and authorization and adds credentials and attributes to outbound Web Services requests.

TransactionMinder also supports the federation of identities in the B2B environment by describing user identities to partners in WS-Security headers, and supporting the consumption of credentials issued by partners, including SAML assertions. The product also allows single sign-on across multiple Web Services used in the same transaction. Please read ZapThink's whitepaper entitled "Context and Identity: The Linchpins of Web Services Security" (WP-0120) for more information on the Netegrity solution.

IV. Conclusions

Web Services are certainly gaining a considerable amount of traction in companies of all industries, sizes, and types. The promise of interoperability and seamless integration across different businesses, applications, and information types is certainly a compelling reason for adoption of Web Services and SOAs for B2B integration. However, the reality of implementing Web Services in B2B integration scenarios requires significant attention to security and reliability issues.

Web Services are standards. And since standards, by their nature, take years before they take hold, the primary means by which companies realize ROI on standards adoption is by cost savings. Typical cost savings is not from internal implementation of standards, but rather in external, B2B-type integration efforts.

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

As a result, standards exhibit the network effect; if few companies implement the standards, there are little cost savings, however as the number of implementers of a standard increase, the resultant cost savings increase dramatically.

While enterprises with significant investments in existing traditional EAI and B2Bi solutions will not be very eager to rip them out and replace them with new, unproven technologies, Web Services are making a significant impact in a number of vertical industries, and as such their use within B2B integration scenarios is all but inevitable. Therefore, it makes sense for these enterprises to make use of emerging technologies, like those provided by Netegrity, to assure that their B2B interactions are secure and reliable.

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com

Copyright, Trademark Notice, and Statement of Opinion

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

ZapThink, LLC • 11 Willow Street, Suite 200 • Waltham, MA 02453 • info@zapthink.com • www.zapthink.com