zapthink research report

Optimizing Web Services in the Enterprise

From Point-to-Point Web Services to Service-Oriented Architectures

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

OPTIMIZING WEB SERVICES IN THE ENTERPRISE

FROM POINT-TO-POINT WEB SERVICES TO SERVICE-ORIENTED ARCHITECTURES

May 2003

Analyst: Ronald Schmelzer

Abstract

XML and Web Services traffic is consuming an increasing percentage of the bandwidth on the corporate network. At the same time, enterprise IT architects and data center administrators are struggling with how to deploy XML and Web Services-based solutions in the enterprise in a reliable, secure, and manageable manner. Furthermore, existing computing infrastructures are increasingly inadequate to meet the demands of high-performance XML and Web Services capabilities.

To solve many of these challenges, vendors are introducing a new category of intelligent network device that are able to intercept, inspect, transform, and redirect XML and Web Services requests according to business policies.

As enterprises move from simple, point-to-point applications of Web Services to building Service-Oriented Architectures (SOAs) that leverage the capabilities of XML and Web Services, end-users must understand how to optimize not only their current XML-based integration solutions, but also the performance of their SOA implementations as they build out these more agile approaches to distributed computing.

Targeted at those who are in charge of deciding how to implement corporate-wide policies for XML and Web Services usage at both the network and application level, this white paper will discuss key issues for effective implementation of XML and Web Services and how they impact different parts of the corporate IT infrastructure,both now and into the future.

Table of Contents

I. The Emerging Class of XML-Based Applications on the Network

No concepts in the software world since the dot.com implosion have received as much attention as the Extensible Markup Language (XML) and Web Services. XML and Web Services are fast becoming a ubiquitous part of enterprise data communications today. XML and Web Services provide a standardized, versatile, and cross-platform way to describe and encode data and simplify intersystem communication. Soon, XML will pervade all aspects of information technology.

The story of XML and Web Services is the story of connecting systems, applications, organizations, and businesses. While many computing technologies have been oriented towards single-system, inter-application communication, such as Microsoft's Object Linking and Embedding (OLE), Enterprise Java Beans (EJB), or ActiveX Data Objects (ADO), Web Services are oriented towards connecting systems of diverse types, including desktop applications, server applications, data sources, or more abstract resources such as organizations, business processes, and businesses. Integration requires interoperability, and interoperability requires the use of standards.

The movement to standards-based computing is imposing new requirements on networks and the applications that ride on top of them.

However, this movement to standards-based computing is imposing new requirements on networks and the applications that ride on top of them. In particular, XML adds considerable requirements for devices on the network that must more deeply inspect network content than they have ever done before. It is these new requirements that are forcing architectural and infrastructure changes in corporate IT environments.

Drivers for XML and Web Services Growth in the Enterprise

The desire to simplify integration is one of the primary drivers for XML adoption.

The desire to simplify integration is one of the primary drivers for XML adoption. The need for integration is spurred by the desire to link multiple systems into a cohesive view of data that can power decision-making, customer interaction, and the delivery of integrated products and services. Matched with the complexity of internal integration is the desire to connect supply chain partners, affiliate networks, distribution channels, and even customers and integrate with a wide array of device types ranging from servers to handheld devices.

Integration is not a simple issue of merely plugging two systems or organizations together. The vision of "plug and play" application and system integration is a pipe dream that may be appropriate some time in the future, but right now, enterprises face the more immediate challenge of connecting arbitrary systems

in a manner that is cost effective, manageable, efficient, and secure. Systems developers and administrators should be able to integrate any system of any type with any other arbitrary system as driven by short-term business requirements, rather than surmount the near-impossible challenge of managing disparate systems in the face of constantly changing business needs.

The main reason why XML is well suited to solving the integration challenge is that it is capable of representing data and processing information in an application neutral, open, and extensible manner. XML allows users to realize a potent solution to today's integration challenges: standardizing data access and encoding among heterogeneous systems. Most current integration solutions encompass a total of at least six systems, and the increasing movement towards greater use of data in real-time enterprise environments, B2B systems, and efforts to make the most of legacy systems is driving a need to integrate with dozens, if not hundreds of systems in a single environment. As the number of connection points increase, so do the complexities and inefficiencies of data transformation, manipulation, and exchange.

Another major driver for XML adoption is the increased need to represent and transmit content that users can repurpose and retransmit to meet different needs. XML enables structured document creation that can then be repurposed for other documents, transformed into different content representation formats, and syndicated to end destinations. As a result, publications can be delivered in "real-time" to end users of the content.

Current Problems with Handling XML and Web Services on the Network

XML is inherently inefficient, insecure, and unreliable.

XML is inherently inefficient, insecure, and unreliable. On top of XML, Web Services add many layers to handle management, process, and security tasks. The result is a data format that consumes a considerable amount of bandwidth, storage, and processor power when compared to binary format or simpler text file format representations of the same information. The reason for this added overhead is simple: XML is a human-readable, platform-neutral, metadata-enhanced, structured, validated format. In addition, using XML-based messages such as SOAP in a transactional, real-time manner may impose more requirements on the system for parsing and processing than the system can handle.

In addition, due to XML's clear text, human readable nature, it is particularly vulnerable to security compromises. By themselves, the core XML and Web Services specifications (SOAP, WSDL, and UDDI) do very little to address security. XML exchanges can span multiple partners, require interaction among many entities, and integrate with corporate security policies for access control and authorization, thus posing additional security challenges. What's worse is that XML documents often encapsulate vital business data that must be protected from prying eyes. As a result, organizations must provide a guaranteed level of security for all XML and Web Services traffic on the network, end-to-end data integrity and confidentiality, and security at the data level.

IT administrators must also insure that all XML-based documents transmitted on the network comply with corporate security policies. As a result, any XML message, including SOAP messages, must be secured using encryption, digital signatures, authentication mechanisms, and privacy controls as appropriate. Systems that process and manipulate the XML documents must add all of these security features without performance degradation while functioning within the overall framework of the network.

XML management tasks will consume increasingly greater amounts of network traffic, processing power, and IT resources.

Finally, systems that rely on XML documents to provide business-critical functionality must be able to manage the quality of service for the delivery of those documents, as well as providing the ability to audit the XML document processing and delivery processes. Since XML by itself does not provide this functionality, XML-enabled systems must supply it instead. The set of features that will rapidly become requirements for IT organizations includes monitoring of XML traffic throughput, quality of service (QoS) measurement, service level agreement (SLA) monitoring, monitoring of long-lived transactions, and billing and metering of Web Services. All of these management tasks will consume increasingly greater amounts of network traffic, processing power, and IT resources.

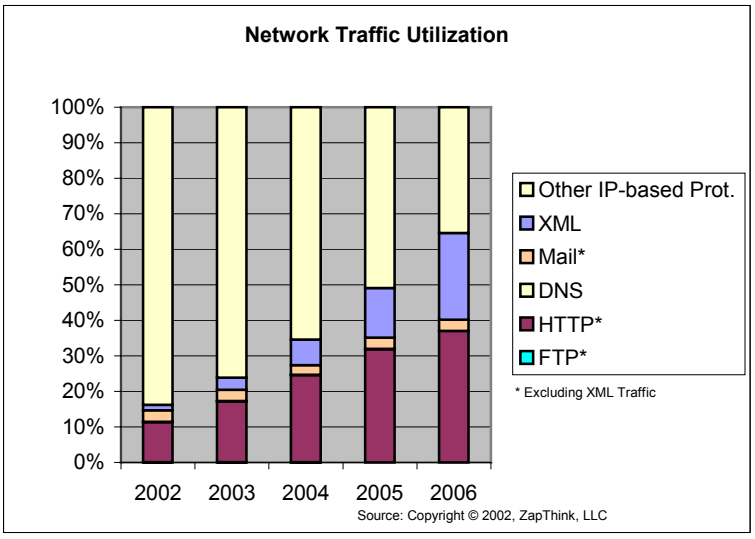
The Growth of XML Traffic on the Network

XML and Web Services are still at the earliest stages of adoption. Right now, enterprises are exploring Web Services as a standards-based interface for integrating systems. However, as companies realize the full architectural benefits of Web Services, there will be a tremendous increase in network traffic for critical XML-based Web Services standards such as SOAP, WSDL, and UDDI. Furthermore, as most software vendors embed Web Services within their software products, XML-based network traffic will increase exponentially.

ZapThink expects XML traffic on the network to greatly increase to just under 25% of all LAN network traffic by 2006.

Consequently, ZapThink expects XML traffic on the network to greatly increase over the next few years. XML represented only 2% of all network traffic on the network in 2002, but this number is expected to increase to just under 25% of all LAN network traffic by 2006. Figure 1 and Table 1 below show the expansion of network traffic over the next few years.

Figure 1: Growth of XML as Percentage of Network Traffic†



(†Other IP-based protocols include instant messaging, file and print sharing protocols, IP-based RPC protocols, P2P formats, and proprietary messaging formats).

Table 1: Percentage utilization of Network Traffic by format

Protocol	2002	2003	2004	2005	2006
FTP	0.06%	0.06%	0.04%	0.03%	0.02%
HTTP	11.33%	17.06%	24.17%	31.42%	36.57%
DNS	0.08%	0.07%	0.06%	0.04%	0.03%
Mail	3.16%	3.81%	4.32%	4.49%	5.18%
XML	1.60%	3.34%	7.09%	13.83%	24.15%
Other	83.73%	75.64%	64.30%	50.17%	35.04%

Source: Copyright © 2003 ZapThink, LLC

As XML and Web Services traffic increases on the network, it will become more difficult for IT and data center administrators to ignore the impact it has on their operations. Just as they have had to deal with the new threats that the Internet and other new technologies introduced to their environment, they will be forced to deal with the new challenges involved in efficiently and effectively handling XML and Web Services traffic.

II. Optimizing Point-to-Point XML and Web Services Implementations

Integration is a critical challenge at the heart of most enterprises' key business systems, including Customer Relationship Management (CRM), Supply Chain Management (SCM), e-Commerce systems, and enterprise portals. Companies experience integration problems once they company installs their second enterprise system. The need for these two systems to communicate forms the basis of the integration challenge.

The need for two or more systems or businesses to communicate forms the basis of the integration challenge.

The simplest integration methodology is "point-to-point," where systems that need to communicate are connected directly to each other. In this scheme, the number of integration or interconnection pathways that must be established grows geometrically (or n-squared) with the number of systems to be integrated. The increasing movement towards business agility and the "real-time enterprise" is driving a need to integrate with dozens, if not hundreds of systems in a single integration environment. As the number of connection points increase, so do the complexities and inefficiencies of data transformation, manipulation, and exchange. We are thus faced with an integration problem that grows at a rapidly increasing rate.

Unfortunately, this integration problem is compounded by the fact that most companies perform integration in an *ad hoc* manner, narrowly focusing on short-term integration needs rather than overall solution effectiveness. The end result is a tangled web of point-to-point integrations that neither meets business requirements nor performs adequately.

Companies need a loosely coupled approach to integration—one that does not require control of systems on both ends nor an intimate connection between the requester of information and the provider.

Using XML and Web Services to Simplify Point-to-Point Integration Challenges

In order to solve the integration cost and complexity issues inherent in traditional, point-to-point integration approaches, companies need a *loosely coupled* approach to integration—one that does not require control of systems on both ends nor an intimate connection between the requester of information and the provider. However, for loose coupling 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 the other systems the company wishes to communicate with. XML represents the

standards-based way of representing information independent of the system type and Web Services provides a standards-based way to connect system interfaces.

Since Web Services, and in particular Service-Oriented Integration (SOI), improves the general approach towards integration, we can expect some significant improvements on ROI over what was possible with traditional approaches to EAI and B2Bi. The improvements to be seen include dramatic reductions in TCO as well as improvements on the tangible and intangible aspects of ROI as follows:

- *Reduction of Cost* – Web Service applications are easier to integrate, utilize lower-cost and more widely available tools, and require less time to create. This reduction plays into the hands of IT managers who have a big investment in existing "legacy" infrastructure and tools that cannot be simply "ripped and replaced."
- *Improvement in Efficiency* – Web Services promise a high degree of reusability, faster time-to-market, and ability to integrate with third-party systems and trusted business parties. Tools offer sophisticated functions to turn existing service implementations into Web Services.
- *Streamline Business Operations* – Web Services enable a common architecture and approach to be pervasive in an enterprise containing heterogeneous legacy systems. Generation facilities streamline the creation of Web Services interfaces, such as the generation of WSDL from existing service implementations or from models.
- *Faster time to market* – Web Services enable companies to become more agile, offering services in increments when they become available, rather than waiting for companies to change or complete whole systems.
- *Potential for New Revenue Streams* – Use of externally supplied Web Services allow maintenance of existing customers or addition of new revenue streams.

Network and Application Implications for Point-to-Point XML Implementations

Enterprises need to optimize not just the process of creating standards-based interfaces to systems, but also the traffic and flow of those standards-based messages.

In order to facilitate point-to-point information exchanges, what is needed is to optimize not just the process of creating standards-based interfaces to systems, but also to optimize the traffic and flow of those standards-based messages. In distributed computing systems that are integrated with a traditional, point-to-point approach, there are fundamentally two types of entities: those that are providing access to functionality (servers or service providers), and those that are requesting functionality (clients or service requesters). In a typical system, service requesters send messages directly to providers, which respond with the results of the requested operation. While the use of XML serves to standardize the end-points and thus reduce the cost of integration, it doesn't change this particular mode of interaction. The direct, point-to-point, client/server approach works in limited volume environments, but does not scale well. For example, if all email systems worked by directly sending email between senders and receivers, the service would fail under high email loads. What is required is either the use of intelligent intermediaries or a change of architecture to change this fundamental reality.

In much the same manner, XML messaging will experience tremendous scalability issues if all communication happens on a point-to-point basis. At a certain threshold, centralizing messaging functions in an XML-aware intermediary greatly improves performance and scalability. Highly scalable distributed systems that are predominantly message-based require flexible buffering of messages and routing, based not only on message parameters such

as origin, destination, and priority but also on the state of the system measured by parameters such as the availability and load of its nodes as well as network traffic information. In this model, it is the role of intelligent intermediaries – software or hardware systems that process XML traffic and Web Services requests between participant end-points – to handle the messaging and operation workload. Based on various message characteristics, these intermediaries can send the message directly to the end point, respond with a pre-cached result, or batch messages to a server for batch response.

Enterprises also need to apply corporate-wide security and performance policies on the XML traffic that flows through its networks. These policies may affect the security, management, performance, and vocabulary features of the XML documents. For example, an enterprise may stipulate that all XML messages bound for outside the network must be compressed, digitally signed, and compliant with their particular industry's specifications. Rather than requiring all XML and Web Services applications to be compliant with this policy, enterprises are looking for a way to centrally apply security, performance enhancement, and transformation rules to selectively controlled traffic.

Finally, users wish to gain increasingly more value out of the XML documents and traffic on the network. As XML becomes an increasingly important part of the corporate IT fabric, IT administrators, managers, and CIOs will want to wring more value out of their XML traffic. For example, they may want more in-depth reporting and auditing of XML traffic, message-tracing facilities, billing and metering functions, and all sorts of value-added XML features that are currently in the imaginations of vendors and end users. Intelligent intermediaries can provide the "tap point" into the XML traffic stream where they can add these functions. Without such an XML-aware intermediary, adding these new features becomes an expensive and custom-coded endeavor.

III. Optimizing Web Services-based Service-Oriented Architectures

Loosely coupled, standards-based Service-oriented architectures enable companies to build agile business processes and IT systems that can respond to change.

By themselves, Web Services are little more than a standards-based way of integrating systems that would otherwise be similarly connected with proprietary interfaces. The application of XML and Web Services to improve the economics of point-to-point integration, however, only scratches the surface of the true potential of Web Services -- enabling companies to build agile business processes and IT systems that can respond to change through the use of loosely coupled, standards-based *Service-oriented architectures*.

Service orientation is an evolutionary approach to distributed computing in which software functionality is made available as business-oriented Services on the network. Open, standards-based Service-oriented architectures built with Web Services form a set of technical principles that can provide companies with the strategic IT direction they need to meet their fundamental business goals. Without such a direction, companies risk losing track of the long-term decisions they must make to remain competitive.

Using SOAs to Achieve Business Agility

Of all the goals that enterprises have today, the one business imperative that Service orientation addresses most directly is the need for *business agility* Business agility is more than simply being able to respond quickly to change; it also means the ability to leverage change for competitive advantage. For businesses to be agile, their technology must be agile as well. Service

orientation, therefore, provides the guidelines for building an IT infrastructure that is agile enough to respond to the needs of the agile business – finally putting business in control of IT, rather than vice-versa.

Information technology is often the area most relevant to discussions of business agility, because achieving agility begins with removing the bottlenecks that impede it, and IT has traditionally been the source of most bottlenecks. In fact, companies are so used to the fact that IT decision-making and implementations impede their organization that technology and its limitations often drive business decisions. Service orientation, however, has the potential to change this equation, and enable business decisions 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.

The difference between the practice of SOA and other approaches to enterprise architecture is in the business agility that SOA offers. For the architect, building an architecture that provides business agility means creating an IT infrastructure that meets as-yet unknown business requirements—a situation that throws traditional IT planning and design out the window. Instead of dealing with concrete requirements from business, SOA considers the next level of abstraction: the ability to respond to changing requirements is the new "meta-requirement." The entire architecture—from the 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.

Network and Application Implications for Web Services-based SOAs

Enterprises must move beyond simple point-to-point applications of Web Services to a broad application of Web Service technologies both within the enterprise and increasingly among business partners.

Forward-looking enterprises are now looking to take the next step and figure out how to leverage the power of Web Services strategically across the enterprise. This next step means moving beyond simple point-to-point applications of Web Services to a broad application of Web Service technologies both within the enterprise and increasingly among business partners. This transition requires more than a simple change in programming practices. This broad application of Web Services technologies requires an architectural change—a move to loosely coupled, standards-based Service-oriented architectures. This new architectural approach requires a different perspective on the role of IT in the organization.

Producers of Web Services that participate in a Service-Oriented Architecture must be able to publish information about them in a service registry, where service consumers can then look up and bind to the services they need. Systems need to expose their capabilities as Services, and other systems that choose to interact with them can simply discover those services and bind to them either at runtime or design-time. Organizations must be able to 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. The enterprise also needs to have a management infrastructure in place that can support the performance of the Services as they are being moved into production as well as once they are live.

IV. XML Proxies: XML-Aware Intermediaries

XML Proxies are applications that monitor network traffic for XML content and perform some activity on that traffic as dictated by business rules.

A class of solutions is emerging to solve all these various challenges in XML-aware network processing. Increasingly, the terms XML Proxy or XML-Aware Intermediary are being applied to these various solutions. XML Proxies are applications that monitor network traffic for XML content and perform some activity on that traffic as dictated by business rules. Specifically, an XML Proxy is a hardware or software solution that actively listens for XML traffic on the network and either passes it along unmodified or performs some action on the XML content.

Most current IP network devices work on the packet level. For example, routers operate by examining TCP/IP packet headers and making network routing decisions based on rules set up for recognizing patterns in those packets. Firewalls work by blocking access to network traffic that doesn't comply with certain allowable parameters, as specified by their source and destination TCP ports (such as port 80 for web traffic and 25 for email) and IP addresses. Load balancing and traffic management systems work by examining Web (HTTP) headers and making decisions based on the URLs and other information contained within. However, the TCP/IP port and HTTP header approaches are too limited for dealing with XML-based content and Web Services traffic. Such traffic has been created to take advantage of standard protocols for transfer, such as HTTP and SMTP, and as a result, the information flows transparently through current network devices. As a result, these devices must become aware not only of the network ports and IP addresses, but also of the content itself that is traveling across the network. Instead of being simply network and IP-aware, these solutions must be *content-aware*. More specifically, they must be XML-aware. They must be able to inspect and understand XML traffic as it flows across the network and perform some sort of activity on the traffic, as policies dictate.

XML Proxies are capable of examining traffic at the content level, and are typically optimized for processing XML and Web Services traffic at the latency required for typical mission-critical business operations.

XML Proxies are capable of examining traffic at the content level, and are typically optimized for processing XML and Web Services traffic at the latency required for typical mission-critical business operations. These solutions must be XML-aware, but are not necessarily specific to any one XML vocabulary (such as SOAP-based Web Services or ebXML). XML Proxies can operate in several different modes, serving as a stand-in for both client and server operations, performing operations itself, or simply allowing the traffic to flow through. In many instances, XML proxies serve as a content-level, XML-aware firewall, protecting external and internal systems from direct contact.

The set of functionality that XML Proxies will provide will increasingly converge on a set of value-added tasks that intermediaries can perform on inbound XML documents prior to being forwarded to end point destinations. While XML Proxies can also pass through XML documents without modification, their real value will be seen as adding one or more of the following pieces of functionality:

- Performance Enhancement
- Security
- Message Routing
- Security
- Performance Enhancement
- Message Transformation
- Monitoring and Management.

XML Proxies can reside anywhere on a corporate network, processing XML traffic as specified by the end points and returning their results to the destination specified by the sender. The main benefit of implementing XML Proxies in this manner is that users can control when and how XML traffic is processed, without involving central IT administrators in purchasing, installation, and maintenance of the system. As the XML Proxy begins to fill a more critical role in the enterprise, a better way to implement the system is as a transparent processing gateway between XML producing and consuming applications. In a transparent system, XML producers do not need to be aware in advance of the presence of the XML Proxy. Rather, they simply create and send XML messages as they normally do. If the message is bound for a destination outside the local network, the XML Proxy will intercept the outbound message, apply business rules, and then forward to the necessary network router for further message forwarding.

There currently exist a growing number of vendors focusing on providing XML Proxy solutions. These vendors roughly are divided into three major groups: hardware vendors focused on providing a data center oriented implementation, software vendors focusing on flexible proxy implementations, and outsourced XML and Web Services networks. In the first case, hardware solutions make use of specialized hardware to drastically improve performance over pure software implementations. Also, IT personnel can easily configure and install hardware-based solutions so that they are ready to install by simply plugging the equipment into the appropriate rack.

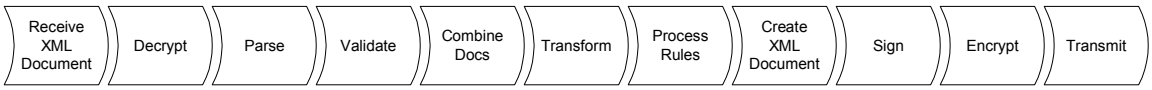
History has shown that there are three four primary drivers to the movement to dedicated hardware devices:

- ***Price*** – As XML processing becomes more of an enterprise requirement, hardware devices can consistently deliver superior price/performance value over software solutions. This is because hardware devices can use specialized components and chipsets and be priced to fit within the budgets of central IT administrative needs rather than as a generic PC system with customized software.
- ***Performance*** – Utilizing specialized hardware, purpose-built devices offer performance that far exceeds those of software solutions. Software implementations require tuning and optimization whereas network devices are designed for plug-and-play applications.
- ***Cost of Ownership*** – Hardware appliances are built so that IT administrators can "set and forget" them. Rather than requiring high level of maintenance as many server-based applications do, purpose-built network appliances have simplified maintenance, administration, and installation interfaces, reducing the overall cost of ownership and maintenance load.
- ***Security*** – Where software is frequently released to the public at large, hardware devices are designed to function as closed environments that provide superior levels of security.

Utilizing specialized hardware, purpose-built devices offer performance that far exceeds those of software solutions.

However, performing XML-aware content inspection is a difficult task. Network devices and application servers are already overloaded handling the basic TCP/IP and HTTP based functions that they were built for. Processing and inspecting XML on the network is computation intensive. Figure 2 below shows just a few of the typical steps that are required to handle each individual XML "packet" on the network:

Figure 2: A Few of the Steps Required to Process XML



Clearly, organizations need efficient ways to process XML and Web Services on the network. While implementing an XML Proxy requires greater participation by corporate IT administrators, the potential benefits far outweigh the costs and risks, especially as XML traffic on the network increases as an overall percentage of total network traffic. In addition, these XML Proxies must fulfill their key goals of providing deep content-level, XML-aware network processing while not sacrificing performance, security, and overall efficiency.

V. F5 Networks: Providing a Platform for Optimized XML and Web Services in the Enterprise

Implementing Web Services and Service-Oriented Architectures that are efficient, secure, and reliable is far from an easy task. What data center administrators and network architects need is a comprehensive, XML-aware solution that address the various different requirements for simplifying and reducing the cost of integrating systems in the enterprise. In essence, there is a "big picture" of Web Services on the network that needs to be implemented. Rather than seeing the various devices and appliances on the network as discrete elements that touch parts of Web Services-based transactions, developers and administrators need to consider the network to be the extension of the enterprise architecture --- effectively making the network a manageable part of the application infrastructure.

Developers and administrators need to consider the network to be the extension of the enterprise architecture.

F5 Networks provides solutions that embody this comprehensive approach to Web Services network management through its BIG-IP, 3DNS, and iControl family of products. The company sees the network as a means to deliver Web Services successfully, and as such the products provide three key benefits to companies currently or looking to implement Web Services and Service-Oriented Architectures:

- Rich interoperability and integration with Web Services infrastructure products and platforms through support for Web Services protocols and direct support for the specific implementation needs of Web Services platforms.
- Providing Web services-based access to network device, effectively providing hardware that looks like and can be controlled like any other software or application, and providing intelligent network control.
- Facilitating and managing Web Services traffic on the network through XML-aware content and message inspection and management.

Focused on solving the last benefit mentioned above, F5 Networks provides an Application Traffic Management solution called BIG-IP that enable enterprises and service providers to optimize the flow of XML-based Web Services traffic. BIG-IP is F5's local area application traffic management solution, that intercepts, inspects, transforms, and directs traffic (including XML-based content) to the correct resource or application within a data center, based upon capacity, performance, or other business criteria. BIG-IP is capable of processing XML

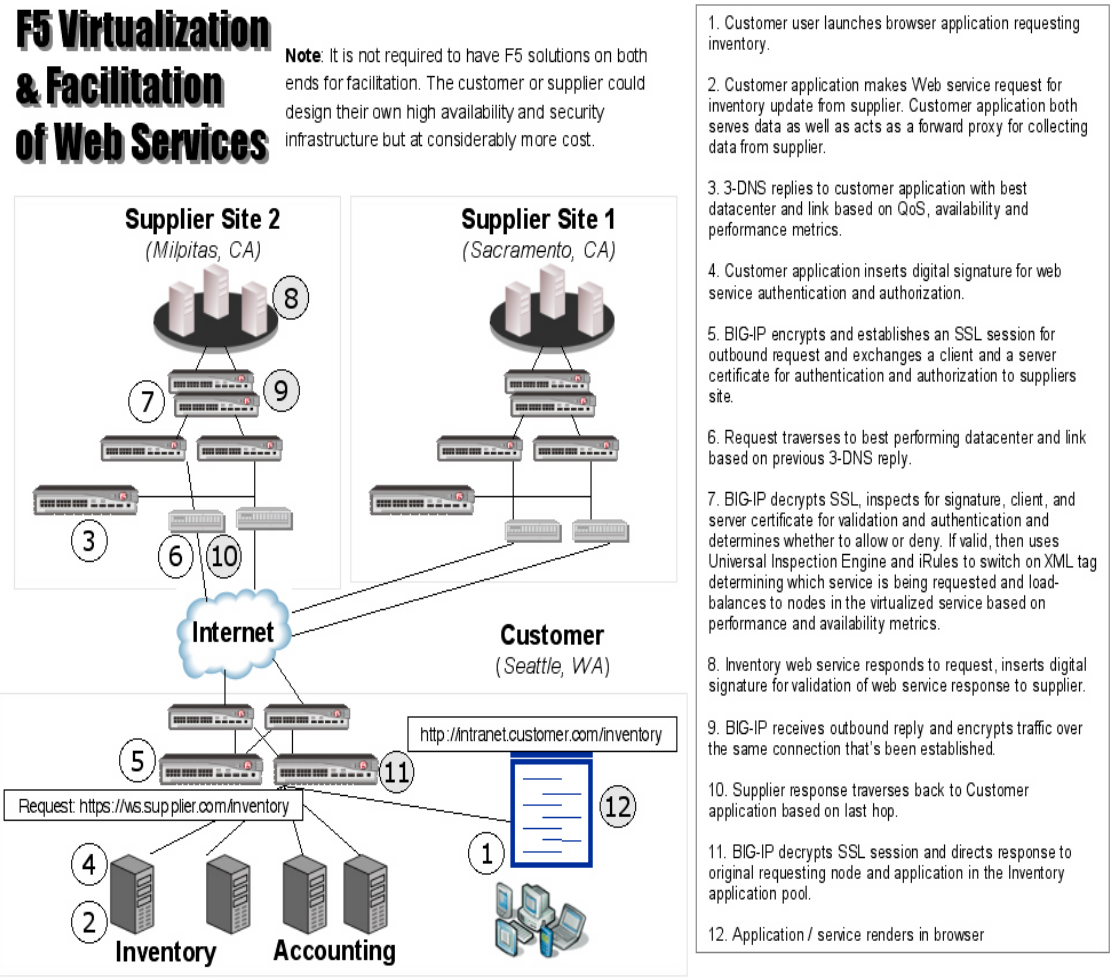
F5's BIG-IP is capable of processing XML message streams at the content level.

message streams at the content level, thus offloading this task from overloaded application servers and handling the reliability, security, management, and scalability requirements outlined earlier in this report. In addition, BIG-IP's flexible Universal Inspection Engine can be applied to inspect for digital signatures and certificates to validate and authenticate requests. BIG-IP's encryption and decryption capabilities not only offload the processor intensive task from the application servers but also streamline and centralize security development.

3-DNS provides wide-area traffic management and high availability of Web Services running across multiple data centers.

The 3-DNS solution furthers the value proposition by optimizing Service-Oriented Architectures and enabling reliable Web Services interaction. The product provides wide-area traffic management and high availability of Web Services running across multiple data centers. The 3-DNS product distributes Web Services requests according to business policies, data center conditions, and network conditions such as round trip time, packet loss and other QOS metrics. Used as an active management layer for Web Services interactions in internal as well as external integration scenarios, 3-DNS ensures global business continuity and Web Service availability, improves performance and reliability through direction of requesters to available Services, and enables traffic control through the setting of policies that determine Service access criteria.

The below diagram illustrates how BIG-IP and 3DNS products can be implemented in a distributed supply-chain scenario:



iControl exposes network equipment functionality as manageable and controllable functionality that can be integrated with application functionality.

Most importantly, F5 Networks has realized that all of its solutions must be as easily accessible on the network as any other application. In this vein, the company has released iControl – a set of interfaces and management technologies that exposes network equipment functionality as manageable and controllable functionality that can be integrated with application functionality as easily as if it was any other application, data, or software functionality on the network. iControl can enable users to create their own interfaces to managing XML-enabled network functionality as well as create entirely new classes of applications that leverage the power of F5 network devices. The iControl API provides access to over 1600 methods using secured, SOAP-based Web Services. Written to be language and platform-neutral, the interfaces can be accessed by a wide range of systems and platforms ranging from desktop and server environments to Web Services-enabled mainframes and handheld devices. The company provides comprehensive support for companies looking to build applications based on iControl, and includes sample applications as well as a sample, graphical Distribution Monitor on the iControl Developer site (www.devcentral.f5.com)

In summary, F5 Network's products transcend the current understanding of what a network device does and its role in the enterprise architecture. By bringing together the aspects of software and hardware control and management of Web Services and Service-Oriented Architectures, F5 Networks is facilitating the optimization of Web Services in the enterprise and helping to make the secure, efficient, and reliable integration of systems a reality.

VI. Conclusions

XML is pervasive. In a matter of years, it will fuel every application, device, and document found in enterprise networks. However, as XML proliferates, it will stress existing systems and enterprise budgets to their breaking points. This is because existing n-tier software architectures and legacy infrastructures were not designed to process this verbose new data type efficiently. What enterprises need is a new way to process XML in the network, rather than in software at the database, application server, or presentation tiers. Yet today's existing network infrastructure is limited to switching lower layer protocols and is unable to detect XML - much less parse and process it.

An emerging class of hardware-based XML-aware network devices addresses the performance, security and management issues that come with XML's use in enterprise applications. These purpose-built network devices enable enterprises to process high volumes of XML in a way that offers high performance, optimal security, lowest total cost of ownership, and greatest centralization of XML processing capabilities

As companies move from using XML and Web Services as a means to simplify point-to-point integration to more complex, but strategic, Service-Oriented Architectures, their requirements on application developers, infrastructures, and networks will likewise increase in complexity. Fortunately, companies are able to realize the benefits of XML and Web Services today, and similarly can find that XML-aware intermediaries not only provide answers for these short-term concerns, but also provide a framework to handle the ever increasing requirements of strategic XML-based SOA implementations as well.

F5 Networks Profile

Profile: F5 Networks	(May 2003)

Date Founded: 1996
Funding: Publicly-held (NASDAQ: FFIV)
CEO / President: John McAdam
Employees: 500
Products:

- BIG-IP
- 3-DNS
- iControl

Address:
401 Elliott Avenue West
Seattle, WA 98119
URL: www.f5.com
Main Phone: 888-88BIGIP
Contact:
Communications Department (206-272-6850)

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