Doc. ID: ZTR-DI101-1
Released: July 26, 2002

zapthink
foundation report

XML Proxies

XML-Aware Network Appliances and Firewalls

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

XML PROXIES

XML-AWARE NETWORK APPLIANCES AND FIREWALLS

July 2002

Analyst: Ronald Schmelzer

Abstract

As the use and proliferation of XML and Web Services spreads throughout the corporate IT environment, so too will the demands on optimizing the performance of the XML data and applying enterprise-wide XML policies. Increasingly organizations are seeking to find solutions that can transparently monitor XML traffic on the network and apply business rules or corporate IT policies such as security, routing, performance, management, transformation, or end-point connection provisioning. Enterprises will implement *XML Proxies*, which can be either hardware Network Appliances,software Proxies, or software Firewalls, as a transparent layer over current LAN and WAN traffic, monitoring and acting on XML data as dictated by pre-configured rules.

Key Points:

✦ Market Overview

- *XML Proxies* are hardware or software solutions that actively listen for XML traffic on the network and either pass it along unmodified or perform some action on the XML content. XML Proxies can operate transparently as XML "gateways" or as auxiliary applications on the network.

✦ Facts & Figures

- ZapThink estimates that XML represents less than 2% of all traffic on the enterprise network in 2002; however, this percentage is expected to increase to almost 25% of all LAN network traffic by 2006.

✦ Analysis

- Current firewall and proxy solutions are inadequate to handle XML traffic. Instead of being simply network protocol-aware, XML Proxies are *XML-aware*.
- XML Proxies are capable of examining traffic at the content level, and can optionally handle other message types such as HTML or EDI.

✦ Future Trends

- XML Proxies will converge on a single set of functionality for handling corporate-wide XML security, management, routing, transformation, and performance enhancement.
- As XML Proxy solutions become increasingly visible in the corporate IT environment, "established" Network Appliance vendors will enter the market.

✦ Decision Points

- XML Proxies can also allow users to implement XML and Web Services solutions without having to constantly modify those applications to comply with various corporate XML policies.
- The increasing need to gain more value out of the XML documents and traffic on the network will drive adoption of XML Proxy solutions.

Table of Contents

I. Report Scope

Increasingly organizations are seeking to find solutions that can transparently monitor XML traffic on the network and apply business rules or corporate IT policies.

As the use and proliferation of XML and Web Services spreads throughout the corporate IT environment, so too will the demands on optimizing the performance of the XML data and applying enterprise-wide XML policies. Increasingly organizations are seeking to find solutions that can transparently monitor XML traffic on the network and apply business rules or corporate IT policies such as security, routing, performance, management, transformation, or end-point connection provisioning. Enterprises will implement these "XML Proxies," which can be either hardware Network Appliances or software Proxies and Firewalls, as a transparent layer over current LAN and WAN traffic, monitoring and acting on XML data as dictated by pre-configured rules.

This report addresses the specific question of what role XML Proxies will perform in the corporate IT environment. In particular, this report covers:

- Definition and taxonomy of XML Proxy solutions including:
  - Hardware XML Network Appliances
  - Software XML Proxies and Firewalls
- Inadequacies of current firewall and network protocol-based switching approaches for handling XML content.
- Differences between hardware and software XML Proxy solutions
- Functionality requirements of XML Proxy solutions
- Drivers for XML Proxy adoption
- Barriers to XML Proxy adoption
- Future Trends and XML Proxy market growth

This report only covers the class of intermediaries that serves to transparently monitor and act on XML traffic over a corporate network. The report specifically does not cover the following topics, although they may be mentioned in the context of discussing XML Proxy approaches:

- Security solutions or platforms (see ZapThink's *XML and Web Services Security* Report [ZTR-WS104])
- Management solutions or platforms (see ZapThink's upcoming report on Web Services Management Solutions)
- Hosted or Managed Network Services
- Service-Oriented Integration, EAI, or B2B Integration solutions (See ZapThink's *Service-Oriented Integration* (SOI) Report [ZTR-WS103])

It is our hope the by reading this report, enterprise users will gain a fundamental understanding of how they can use XML Proxies to optimize performance and utilization of XML and Web Services in the enterprise. Vendors who read this report will also be able to gain an understanding of the emerging XML Proxy market and the role that Network Appliance and Software Proxy solution vendors can play.

II. The Role of the XML Proxy

XML is creating new exposures–security, performance, and otherwise–that corporate IT environments must appropriately deal with.

XML is creating new exposures--security, performance, and otherwise--that corporate IT environments must appropriately deal with. In particular:

- *XML is inefficient* – XML messages consume significantly more bandwidth than other protocols and message formats. This verbosity is

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

primarily due to the fact that XML is a text-based, human-readable, metadata-encoded language. A SOAP message will be substantially larger than a corresponding binary-format RPC call.

- *XML is insecure* – Due to XML's clear text, human readable nature, it is particularly vulnerable to security compromises. As a result, any XML message, including SOAP messages, must be enhanced with security features including encryption, digital signatures, authentication mechanisms, and privacy controls. These features will further add to an XML message's bulkiness.
- *XML transformation is processor intensive*– XML messages that are destined for a presentation format such as HTML, WAP, or some another XML format will need to be transformed using XSL or similar methods. In particular, web sites that are XML based will require XSL transformation on the fly, increasing parsing and processing loads on servers.
- *XML Messaging formats must be understood by intermediaries* – XML by itself is just a document format, and not a messaging protocol. SOAP, ebXML, and other specifications add routing and other messaging capabilities to the format. Intermediaries will need to understand these messaging formats and therefore must be able to parse, process, and understand the specific routing requirements of different XML vocabularies.

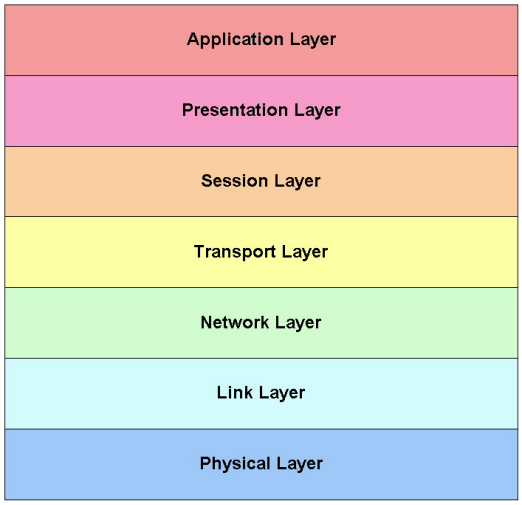
In order to process high volumes of XML content on the network, hardware and software devices must be able to understand not only network protocols, but also the XML-based content traveling on these protocols.

In order to process high volumes of XML content on the network, hardware and software devices must able to understand not only network protocols, but also the XML-based content traveling on these protocols. As we will show in this section, current network technologies are not capable of meeting this demand, thus requiring a new category of XML-aware application.

2.1 The Evolution of Networking Devices and Applications

Over the past two decades, the market for network equipment and software of all kinds has evolved from simple point-to-point, proprietary, and often slow protocols to a highly structured universe of networking technologies that can facilitate communications at many gigabits per second. Over the course of this evolution, the Open System Interconnection (OSI) model for networking architectures has been increasingly accepted as the way to understand the various different "layers" of networking technology that enterprises must implement to facilitate communications between different systems.

Figure 2.1: OSI Network Model



As shown, above, the OSI model encapsulates seven basic levels or layers:

- *Physical* – Defines the physical medium itself for communication, such as thinnet, thicknet, unshielded twisted pairs (UTP), fiber, etc. All media are functionally equivalent at this level, but the main difference lies in bandwidth capacity, convenience and cost of installation, and maintenance.
- *Data Link* –Defines the format of the data that is transmitted on the physical medium. Includes a representation for a network data frame (or packet), checksum, source and destination address, and actual data to be transmitted.
- *Network* – Responsible for directing data packets over the network and to other networks. The network layer may have to break up large packets for later assembly, and also controls how network systems or "nodes" can locate others on the network.
- *Transport* – Controls how transmissions occur and how systems handle them. Transport protocols establish connections between hosts on the network as identified at the Network layer. Transport protocols also keep track of packet delivery order, timeouts, and resend requirements.
- *Session* – Establishes, manages and terminates connections between applications. RPC calls over TCP/IP represent a type of session, as does HTTP and SMTP.
- *Presentation* – The presentation layer converts local representation of data to a canonical form and vice versa.
- *Application* – Supports application and end-user processes. The application layer identifies communication partners and quality of service. In addition, the application layer addresses user authentication and privacy.

Vendor Focus

3Com
Cisco
Linksys
Intel
Nortel

As networking technology has evolved, so has the gravitation towards specific technologies in each of the first few layers: Twisted-pair cable at the Physical layer, Ethernet at the Link layer, IP at the Network layer, and TCP at the transport layer.

At the first four levels of the OSI stack, vendors such as **Cisco**, **Nortel**, **3Com**, **Linksys, Intel,** and dozens of others have produced hardware network appliances

that sought to streamline and improve the efficiency of handling network traffic. These vendors have developed a series of products that go by a variety of terms:

- ***Gateway*** — Software or hardware that enables communication between computer networks that use different communications protocols.
- ***Router*** – A device in a network that handles message transfer between computers and networks.
- ***Switch*** — A network device that selects a path or circuit for sending data traffic to its destination. Switches may also determine the route of data on a network. In general, a switch is a simpler and faster than a router, which requires knowledge about the network to determine routing.
- ***Proxy*** — A system that is authorized to act on behalf of another system, although may simply pass traffic through if so instructed.
- ***Firewall*** – Any of a number of security schemes that prevent unauthorized users from gaining access to a computer network or that monitor transfers of information to and from the network.
- ***Hub*** -- A network device that connects multiple devices on a network so that they can communicate with each other. Hubs are often simpler and less efficient than switches and control traffic on a small segment of a network.
- ***Bridge*** – Synonymous with gateway, router, and/or switch.

These terms have all melded together in different ways. In effect, they are all "Network Intermediaries" in that they facilitate communications between systems, but aren't the end systems themselves. A general definition of intermediary is: "*Acting as a mediator or an agent between persons or things.*" In essence, all of the devices above function as intermediaries for different layers of the OSI stack.

Most network device terms have lost their specific meaning when used beyond the third layer of the network stack.

As a result, most of these terms have lost their specific meaning when used beyond layer 3 of the network stack. In fact, the word "switch" has been alternately equated with gateway, router, hub, and bridge. As a result, it is becoming harder to determine network device functionality simply based on the term applied. Are certain vendors' "firewall" offerings really "routers" or "switches"? Users must look beyond the name of such devices and determine each device's true functionality to determine how best to use them on the network.

2.2 Why Current Network Protocol-based Solutions Are Not Adequate to handle XML Traffic

At higher levels of the stack, the difference between the various levels are fuzzy. While the core protocols for Internet applications such as HTTP, SMTP, and FTP clearly operate at a lower layer than the applications that run on top of them, at what layer of the stack do the emerging XML and Web Services protocols such as SOAP, WSDL, and UDDI operate? This is difficult to ascertain since these are primarily *content-oriented,* rather than *protocol-oriented* specifications. While a case can be made that SOAP is a message format, the OSI model is inadequate to describe the nature of most XML formats. While the OSI model may still be a mechanism to assist in the understanding of traditional networking products, new networking products must transcend the limited OSI model and focus not on the message packet or envelope, but rather the content of the message itself.

Decision Point

New networking products must transcend the limited OSI model and focus not on the message packet or envelope, but rather the content of the message itself.

In particular, the TCP/IP-based approach (level 3 and 4) of current proxy and firewall solutions for managing network traffic misses the mark for handling XML traffic. As defined above, a firewall is a system or group of systems that enforces

an access-control policy between two networks. It can be used to protect a trusted network from an untrusted network, and as such consists of two operations: one to block the traffic, and the other to permit traffic. These traditional firewalls apply security mechanisms such as Secure Sockets Layer (SSL), Transport Layer Security (TLS), or IP Security (IPSec) at the network and transport levels of the OSI stack.

Current TCP/IP-based firewall and router solutions can be put to use in a corporate network by presenting just a single point of access to the outside world, thus hiding the real structure of the corporate network from intruders. In the process of accomplishing this task, these firewall solutions can usually provide full auditing and reporting facilities. As network protocol-based solutions become more of a commodity, they are becoming more complex and functionality rich, blurring the line between firewall, switch, and router. Some boxes handle such wide-ranging functionality such as network routing, bandwidth control, virus scanning, and network security. Clearly, there has become a market for a general "TCP/IP box" that will handle all these disparate packet-level functions.

Most current firewalls work by blocking access to all network traffic, except ones that run on certain TCP ports such as web traffic (HTTP port 80 and HTTPS port 443) and email traffic (SMTP port 25). Firewalls also can ban all network access by hosts at certain IP addresses or based on certain usage characteristics. These firewalls operate by either static packet filtering (making simple deny or permit choices depending on the network address of the packet) which is cheap and fast, or Dynamic Packet Filtering/Stateful Inspection, which makes its decisions based on all the data in the packet.

Proxy Server firewalls act as an intermediary for user

requests, setting up a second connection to the desired resource. The proxy "stands in" for both client and server operations, performing operations or simply allowing the traffic to flow through. However, in many instances, the proxy serves as a firewall, protecting external and internal systems from direct contact.

The TCP/IP port model is too simplistic for dealing with XML-based content and Web Services traffic in particular.

However, the TCP/IP port model is too simplistic for dealing with XML-based content and Web Services traffic in particular. While Stateful Inspection based systems to some extent monitor packet traffic, most Web Services run over standard network ports such as HTTP and HTTPS. However, TCP/IP-based firewalls are programmed to permit or deny all traffic through these ports, and XML/Web Services traffic might be undesirable at the content-level, unlike standard HTTP traffic. As such, firewalls will need to become aware not only of the network ports and IP addresses, but also of the content itself that is traveling across the network. In this regard, current firewall, router, proxy, and switch solutions are inadequate. Instead of being simply network and IP-aware, these solutions need to be *content-aware*. More specifically, they need to be XML-aware. They need to 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.

2.2.1 The term "XML Application Firewall" Is Insufficient and Incorrect

Recently, a leading analyst firm has proposed a new term for XML-aware firewalls, proxies, and switches: *XML Application Firewalls*. They have used the term in passing to refer to firewalls and proxies that reside at the application layer of the network stack. However, this term is insufficient for describing many of the features of emerging XML-aware Intermediaries. Also, in many ways, this term is inaccurate in describing the functionality of such XML-aware intermediaries.

In many ways, this term "XML Application Firewall" is inaccurate in describing the functionality of XML-aware intermediaries.

The first problem with this phrase is the current understanding of the meaning of the term "firewall." There are a number of definitions of the firewall, but perhaps the more straightforward is "a mechanism used to protect a trusted network from an untrusted network." Therefore, a firewall is a hardware or software solution that is meant to enforce access control policies between two networks, and thus is essentially an implementation of policy. While firewalls can also route and process network traffic, their primary function is to keep networks separate. However, this function is clearly not the primary focus for XML-aware intermediaries that actually focus on performance, transformation, and management .

The problem with using the word "application" in "application firewall" is that the application layer is too broadly defined. As defined earlier, the application layer supports application and end-user processes such as identification of communication partners, management of quality of service, provision of user authentication and privacy features, and specification of constraints on data syntax. Everything at this layer is application-specific. This layer provides application services for file transfers, e-mail, and other network software services. Telnet and FTP are applications that exist entirely in the application level. Tiered application architectures are part of this layer. If all of Telnet, FTP, email, etc. are part of the application layer, then the content that goes over such applications is really just a small part of the application layer or can be considered to be a level above the application level (the highest level on the OSI model). As a result, "XML application firewall" is a term whose vagueness leaves much to be desired. XML-aware intermediaries focus on the content of the message rather than the message itself. As such, the term "application" is confusing.

2.3 XML Proxies

A better term to describe the evolving role of XML-aware intermediaries is "XML Proxy." XML Proxies describe applications that monitor network traffic for XML content and performs some activity on that traffic as dictated by business rules. In many ways, this coincides with the definition of the term "proxy" as defined above. While not a perfect term, XML Proxy can be defined as follows:

> **XML Proxy:** Hardware or software solutions that actively listen for XML traffic on the network and either pass it along unmodified or perform some action on the XML content. XML Proxies can operate transparently as XML "gateways" or as auxiliary applications on the network.

Decision Point

XML Proxies are capable of examining traffic at the content level, and can optionally handle other document types such as HTML or EDI content.

XML Proxies are capable of examining traffic at the content level, and can optionally handle other document types such as HTML or EDI content. They must be XML-aware, but are not necessarily specific to any one XML vocabulary (such as Web Services or ebXML). We distinguish XML Proxies from other types of network intermediaries in that they are XML-aware, while others may be TCP/IP aware or HTTP aware. However, other than the fact that XML Proxies facilitate (intermediate) XML communications and are capable of processing XML documents, the role that XML Proxies fill varies based on the activity they perform on the XML content.

While the term "XML-aware Intermediaries" is more inclusive and generally a more accurate assessment of the current products and services on the market, ZapThink believes that "XML Proxy" is a more marketable name, and hence one that can more adequately distinguish the current class of products and solutions on the market.

2.4 Use and Context of XML Proxies

There are two primary ways that XML Proxies can be used in a corporate IT environment:

- In an explicit fashion, where communicating end points specify the XML Proxy to process their traffic.
- In a transparent fashion, where the XML Proxy intercepts XML traffic bound to an end destination, performs some operations based on business rules, and then forwards the newly reconstituted message to the end destination.

2.4.1 Explicit XML Proxies

The main benefit of implementing XML Proxies in an explicit manner is that users can control when and how XML traffic is processed, without needing to involve central IT administrators.

In the first case, 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. However, implementing XML Proxies in this explicit fashion can be cumbersome and inefficient as developers of XML applications must configure their applications in advance to communicate with the XML Proxy. If the location of the XML Proxy changes, or if it is temporarily unavailable, then the developer will need to reconfigure the application. Furthermore, in an explicitly-defined XML Proxy configuration, outbound XML

traffic must be forwarded to other devices and systems if it is to be routed outside the network, resulting in operational inefficiencies.

2.4.2 Transparent XML Proxies

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 this way, the enterprise can realize all the benefits of an XML-aware intermediary. While implementing an XML-aware intermediary 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 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.

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.

2.4.3 SOAP Intermediaries

One of the challenges facing enterprises who use transparent XML Proxies is that such Proxies will not intercept XML messages bound for the local network. Since the application developer has not coded the application with the XML Proxy in mind, messages will simply be forwarded to their local destination without any value-add being performed by the XML Proxy. One of the possible ways to circumvent this behavior for applications that use Web Services is by implementing a so far little-used feature of SOAP: the use of SOAP intermediaries.

As defined by the W3C, SOAP intermediaries are applications that can process parts of a SOAP message as it travels from its origination point to its final destination point. In a typical Web Services implementation, the assumption is made that a SOAP message originates at an Web Services provider and is sent to Services consumers via zero or more SOAP intermediaries. Within a SOAP message, users may request that the message be forwarded to another node on behalf of the initiator of the message.

The SOAP header carries the instructions for SOAP intermediaries, rather than the body of the SOAP message that carries the message payload. This fact does not mean that an intermediary cannot look at, process, or change the SOAP message body; it certainly can do that. However, SOAP itself does not provide a mechanism for instructing intermediaries to inspect the content of the SOAP message body. Instead, the SOAP header's regularly formatted structure provides a mechanism to supply information for SOAP processing end points, including intermediaries. Specifically, SOAP specifies the "SOAP-ENV:actor" attribute, the value of which is a URI that identifies who should handle the header entry. The absence of this attribute implies that only the final destination should process the header.

SOAP defines two different types of intermediaries: forwarding intermediaries and active intermediaries. Forwarding intermediaries process the message according to the SOAP processing model and relay the message to its end destination without really taking any action on the SOAP message itself. Forwarding intermediaries follow rules that specify how to construct the forwarded message, such as the placement of inserted or reinserted blocks.

Active intermediaries provide the same processing capabilities as forwarding intermediaries, but also provide additional functionality that may modify the outbound message in ways not described in the inbound message. For example, these services may provide logging, security, content modification and tracing. In many ways, the vision for an Active SOAP Intermediary is the same as that for the XML Proxy, although the XML Proxy can deal with more than just SOAP traffic and can perform additional operations on the XML not specified by SOAP intermediaries. However, the use of SOAP intermediaries can provide a better means for inserting XML Proxies into the Web Services exchange process in a transparent manner than relying on the XML Proxy to intercept all XML traffic destined for the local network.

Decision Point

The use of SOAP intermediaries can provide an optimal means for inserting XML Proxies into the Web Services exchange process in a transparent manner

One of the other benefits of using SOAP intermediaries is that there need be no requirement that the complete message path be known at the time it leaves the initial sender. Rather, the XML Proxy (functioning as SOAP Intermediary) can make the best decision based on business rules and other context-aware dependencies.

III. XML Proxy Functionality

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.

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:

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

3.1 Security

As mentioned earlier, XML documents have no inherent sense of security. What's worse is that XML document often encapsulate vital business data that must be protected from prying eyes. While a number of major security solutions exist to protect XML and Web Services transactions (see ZapThink's *XML and Web Services Security* Report [ZTR-WS104]), XML Proxies can help to establish security policies in a comprehensive manner, assuring that all XML documents transmitted on the network comply with corporate security policies. XML Proxies can monitor XML traffic for specific patterns and apply content-level security features such as encryption down to the element level, digital signatures, and authorization features.

Current TCP/IP-based firewalls are wholly inadequate at providing content-level 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. As a result, XML-aware intermediaries must provide:

- A guaranteed level of security for all XML traffic on the network
- End-to-end data integrity and confidentiality
- Security at the data-level.

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

XML Proxies can work with other security solutions by applying different security technologies based on the destination or nature of the XML documents. For example, Web Services messages bound for external networks may need to have all of their data encrypted, while those messages bound for internal destinations can simply have the appropriate digital signatures. Also, XML Proxies can handle ebXML, RosettaNet, and other XML vocabularies in different manners, applying the appropriate security capabilities as appropriate. One benefit of an XML Proxy solution is that it can be the "last toll booth" of security, assuring that no messages cross the enterprise firewall without meeting minimum requirements. Enterprises will still need additional, specific security solutions, since application developers can always choose to implement their own level of security behind the firewall. What the XML Proxy adds is an additional safety layer so that otherwise insecure applications can still be compliant with enterprise security policies in the IT environment.

One of the interesting security challenges in an XML Proxy environment is that point-to-point encryption protocols such as Secure Sockets Layer (SSL), Transport Layer Security (TLS) or IP Security (IPSec) must be "terminated" by the intermediary device and reinitiated for the outbound connection. In effect, the intermediary must establish secure connections with both ends of the communication pipe, potentially becoming a bottleneck or a point of security compromise.

XML Proxies can implement the latest XML security technologies in a transparent manner.

Also, XML Proxies can implement the latest XML security technologies in a transparent manner, thus eliminating the need for developers to apply new technologies or patch existing implementations in order to be compliant with the latest standards. XML Proxies can therefore provide a "shell of security" over existing applications, sheltering developers from the need to keep track of or even implement XML security or privacy technologies. A number of hardware and software XML Proxy vendors have focused initially on security functions including **Forum Systems**, **Sarvega,** and **Vordel**.

Vendor Focus

Forum Systems
Sarvega
Vordel

3.2 Performance: Compression and Caching

Another major challenge with XML is its verbosity and inefficiency. XML is equally intended for machine processing and human readability. The combination of these two factors results in message sizes that are easily 10 to 50 times larger than equivalent purpose-built, binary protocols. As XML usage expands in the network, network bandwidth will increasingly be eaten up by inefficient XML traffic. At some point, IT administrators will demand more effective use of network resources.

Decision Point

There are two primary means to improve the performance of XML on the network: caching and compression.

There are two primary means to improve the performance of XML on the network: caching of XML and Web Services requests and compression of the XML traffic itself. In the first case, the XML Proxy can serve a vital role in eliminating redundant requests on systems. There are a few major benefits of XML document caching:

- Reduce network utilization
- Reduce load on Web Services delivery platforms
- Improve services reliability by accommodating sporadic network failures
- Provide testing of Web Services functions without impacting systems in production.

Of course, caching is a technically difficult task, requiring a complex mix of predictive technologies and storage solutions to make sure that the results the cache server delivers to the requester are absolutely correct and current. Caches

can only handle some percentage of all services requests, and forward the remainder to the originating provider.

However, compressing the XML traffic into a binary format can squeeze out some additional performance from these operations as well. Since XML is a text-based, highly redundant data format, XML message senders can employ widely used compression schemes such as gzip to achieve up to 90% compression of XML data streams. While message compression may seem trivial to implement, the two major challenges to practical adoption of compressed XML is that both sides of the communication path must be able to understand and process the document, and the processing of compressed documents involves processor overhead and latency that might not be acceptable in high volume environments, specifically where the network needs compression techniques most. In this vein, hardware-based XML Proxies might have the most to offer for both compression and caching operations.

3.3 Monitoring and Management

Since XML traffic will be flowing through the XML Proxy, it only makes sense for the device to monitor the traffic. As covered in ZapThink's upcoming *XML and Web Services Management* report, the major components of Web Services management include:

- Monitoring of XML traffic throughput
- Quality of Service measurement
- Service Level Agreement (SLA) monitoring
- Monitoring of long-lived transactions
- Billing and metering

As XML-aware intermediaries, XML Proxies can be inserted into the XML stream to provide monitoring and management functionality.

As XML-aware intermediaries, XML Proxies can be inserted into the XML stream to provide the above functionality. As users crave more information about what is happening to XML and Web Services traffic that circulates inside the corporate network or crosses the corporate firewall, XML Proxies must be able either to provide hooks to XML and Web Services management utilities or provide some basic management capabilities themselves. As Web Services Management becomes a sustainable market in itself, it will increasingly make more sense for XML Proxies to simply provide hooks to these applications rather than advanced management functionality themselves.

3.4 Routing

As a natural extension their role as an XML-aware intermediary, XML Proxies must route XML information that is inbound to the device to a final destination. XML Proxies can serve to add intelligence to this function by determining the best route to the desired XML destination, or they can deny service to malformed, inappropriate, unauthorized, or malicious requests. In this manner, XML Proxies function much like firewalls, except that they can also follow business rules that can make intelligent routing decisions based on the content of the XML message, rather than on packet-based parameters. XML Proxies can thus be used in conjunction with traditional firewall applications, by analyzing traffic that successfully passes through the firewall on a content level.

3.5 Transformation

Another key activity that XML Proxies can perform is the transformation of an XML message to other formats such as alternate XML representations (from one vocabulary to another), EDI, or other text formats. Since the XML Proxy serves as

an XML-aware intermediary, the IT manager can configure the Proxy with business rules that not only determine the final destination routing, but also the format that the destination accepts. The XML Proxy can then transform the original document into its final format, using XSL or other technologies, and then transport it to its final destination using the protocol as defined by the business rules. XML Proxies can also traverse protocol boundaries--accepting XML requests in one protocol format (say, HTTP) and spitting it out in another (say, SMTP and FTP). For example, an XML proxy can deliver an inbound SOAP packet delivered over HTTP to an endpoint using EDI over FTP.

XML Proxies can also traverse protocol boundaries--accepting XML requests in one protocol format and spitting it out in another.

Transformation is a resource intensive and technically complex activity. While XML-to-XML conversions can simply use XSLT, the transformation of XML to other formats is more like a "mapping" function that requires not only a sophisticated transformation engine, such as that offered by dedicated transformation tools offered by **XML Global** and **Contivo**, but also a visual user interface to assist in the creation, deployment, and management of mappings. One hardware XML Proxy vendor, **DataPower**, has focused primarily on these transformation issues.

Vendor Focus

Contivo
DataPower
XML Global

IV. XML Proxy Solutions

Decision Point

There are two primary ways that enterprises can implement XML Proxy solutions: in dedicated hardware Network Appliances or in specialized software Proxies.

There are two primary ways that enterprises can implement XML Proxy solutions: in dedicated hardware (XML Network Appliances) or in specialized software (Software XML Proxies). These approaches are not mutually exclusive--there are distinct scenarios in which hardware implementations are most appropriate, other scenarios where software is best, and yet others where combinations of hardware and software solutions are optimal. This section of the report aims to explore the various solutions available and the scenarios where they are most applicable.

4.1 Hardware XML Network Appliances

There exist a rapidly growing segment of vendors who are applying dedicated hardware technology for XML Proxy solutions. These vendors are creating specialized software and hardware combinations that can fit within traditional IT rack environments. There are a number of scenarios where XML Network Appliances are the most appropriate implementation of XML Proxy solutions:

- *Performance* – Network Appliance solutions can make use of specialized hardware and optimize their software solutions to make use of this hardware to drastically improve performance over pure software XML Proxies. In particular, there exist a variety of hardware solutions for accelerating encryption, parsing, and transformation operations.
- *Controlled Installation Environment* – IT personnel can preconfigure Network Appliance solutions so that they are ready to install by simply plugging the equipment into the appropriate rack. As a result, IT shops can control their installation process, making sure that they have properly configured certain security, routing, transformation, and management features prior to installation.
- *Centralized installation in IT environment* – IT personnel can install Network Appliances in a central network operations environment, allowing developers to pass off device administration and maintenance responsibilities to IT administrators. In addition, administrators can manage Network Appliances as they manage other devices on the network rack, with technologies such as SNMP, and provide backup, power, cooling, and appropriate equipment facilities.

- *Different purchasing cycle* – IT managers purchase and account for Network Appliances differently than software. They can be depreciated over many years, and can fit into global IT purchasing budgets in addition to divisional or departmental IT budgets. This allows greater flexibility in selling Network Appliance solutions by offering a choice of customer: either the application developer or the central IT administrator.
- *Simple channel sale* – Reseller channels that currently sell network equipment and associated products can easily sell Network Appliances as well. Online and retail stores can sell these devices, as well as value-added channels such as VARs and Systems Integrators.

Vendor Focus

DataPower
Forum Systems
Sarvega

Currently, vendors such as **DataPower**, **Forum Systems**, and **Sarvega** are offering XML Network Appliances focused at providing XML Proxy solutions. However, these three vendors offer slightly different sets of application functionality. DataPower has an XML transformation-oriented appliance, while Forum Systems is security-oriented, and Sarvega is both security and transformation-oriented. Since each of these systems offers different capabilities, end users seeking to evaluate XML Network Appliance solutions for XML Proxy functionality in 2002 should first seek to establish which features they are interested in before evaluating appliances. ZapThink expects that these vendors will increasingly converge on the functionality identified in this report over the next 12-24 months.

4.2 Software XML Proxies

Another key option for implementing XML Proxies is to implement software solutions that reside on servers on the network. These software solutions provide a flexible mechanism for processing XML traffic without having to purchase dedicated Network Appliances. In addition, these solutions can leverage application servers and other Web Services execution environments by residing on the same server, providing a greater level of integration with XML creation environment.

- *Installation Flexibility* – IT personnel can install Software XML Proxies in a variety of ways, including on the same machine as existing application servers, on department or division-level servers, or as a separate test or pre-deployment system. Software systems, as opposed to Network Appliances, leave their implementation details up to the user, allowing for different hardware choices for scalability and robustness.
- *No need to get central IT involved* – Since implementation choices are left to the user, only department or division level IT personnel must be involved in the purchase and installation process. Therefore, the sales cycle need not be elongated by involving central IT or capital acquisitions departments.
- *Easily OEM-able* – Software vendors who wish to OEM XML Proxy functionality can easily adopt Software XML Proxies . This OEM scenario becomes increasingly likely as vendors seek to add additional value-add to XML-aware intermediaries.

Vendor Focus

Actional
Flamenco Networks
Primordial
Reactivity
Vordel
Westbridge Technology

Just as in the case of Network Appliances, there are a number of different vendors offering software XML Proxy solutions from different viewpoints. Focusing on security are **Reactivity** and **Vordel. Westbridge Technology** focuses on security and management, while **Actional** and **Primordial** offer those features plus routing and transformation capabilities. **Flamenco Networks** focuses mainly on routing and end-point provisioning. However, as discussed above in the Network Appliance section, each of these vendors will increasingly converge on

the set of functionality features that will be common to all XML Proxy applications: security, performance enhancement, management, routing, and transformation.

4.3 "Cross-over" Software and Hardware Solutions

ZapThink sees an increasing trend towards traditional firewall and network applications "crossing-over" into the XML Proxy market.

In addition to focused XML Proxy solutions we described above, a number of other vendors are starting to XML-enable their product lines. ZapThink sees an increasing trend towards traditional firewall and network applications "crossing-over" into the XML Proxy market. While many of these solutions offer basic functionality for XML and Web Services communication, they will increasingly make their presence felt in the market.

Vendor Focus

AmberPoint
CheckPoint
Cisco
Intel
KaVaDo
Nortel
Sanctum
Stratum8

In particular, ZapThink is seeing increased interest in XML-aware intermediaries from companies such as the traditional hardware and software firewall and proxy vendors including **Cisco**, **Intel**, **Nortel**, and **CheckPoint.** These vendors, among many others, are slowly adding content-level features capable of handling some of the requirements of XML Proxies. In addition to these vendors, security software vendors such as **KaVaDo, Sanctum,** and **Stratum8** are adding the capability of securing, monitoring, and managing XML content such as Web Services operations with limited XML Proxy features. Web Services management vendors such as **AmberPoint** are also increasingly looking to provide XML Proxy-like features. As these solutions become full-fledged XML Proxy vendors, ZapThink will track their capabilities. It is expected that as these vendors become XML Proxies, their functionality will converge with the XML-aware Network Appliances and software XML Proxies.

V. Drivers for XML Proxy Adoption

There are a number of business drivers that are spurring the adoption of XML Proxy solutions. As discussed in detail below, these drivers include:

- The need to manage increased volume of XML network traffic.
- The desire to establish a consistent XML usage policy across the enterprise
- The desire to simplify external (B2B) integration
- The wish to increase the value of the XML and Web Services on the network.

5.1 Managing increased volume of XML traffic on the network

One of the major drivers for the adoption of XML Proxies is that as XML traffic increases on the network, enterprises will need specialized applications to achieve the desired economies of scale. In distributed systems, there are fundamentally two types of entities: those that are providing functionality (servers or service providers), and those that are requesting functionality (clients or service requesters). In a typical non-proxy system, service requesters send messages directly to providers, which respond with the results of the requested operation. While this client/server approach works in limited volume environments, the direct request-response model 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. Email servers function as aggregation points for inbound and outbound messages, optimizing the network flow and handling service outages.

Decision Point

Centralizing messaging functions in an XML-aware intermediary such as the XML Proxy can greatly improve performance and scalability.

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 such as the XML Proxy 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 transparent intermediaries to handle the messaging and operation workload. Based on various message characteristics, the XML Proxy can send the message directly to the end point, respond with a pre-cached result, or batch messages to a server for batch response.

5.2 Enforcing corporate XML policies and normalizing XML implementations

XML Proxies can also allow users to implement XML and Web Services solutions without having to frequently modify those applications to comply with various corporate XML policies. 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 ebXML specifications. Rather than recoding all XML and Web Services applications to be compliant with this policy, the XML Proxy can apply security, performance enhancement, and transformation rules to all outbound-only traffic, while leaving all behind-the-firewall traffic alone in its original format. In this way, XML Proxies can not only help applications comply with corporate XML policies, but can also bring different XML and Web Services implementations into conformance with a single XML methodology.

5.3 Simplifying external XML integration

Another major driver to implementation of XML Proxies is the need to simplify the process of integrating with external partners. Since XML Proxies can normalize a company's XML "footprint" to the outside world, they can be used very effectively to help reduce the amount of time that it takes to communicate with new business partners. In effect, XML Proxies act as B2B integration applications, however, without the complex business process management and semantic integration capabilities. Naturally, more sophisticated integration-specific applications such as those offered by Service-Oriented Integration (SOI) techniques must handle these capabilities (see ZapThink's *Service-Oriented Integration* Report [ZTR-WS103]). However, XML Proxies can help to simplify the process of exposing SOI solutions to actual end points. Since XML Proxies can accept XML traffic from multiple SOI solutions and XML-producing applications, they can serve as aggregation points for XML traffic and thus simplify external XML integration.

5.4 Providing Value-Added Services for XML

A major driver for adoption of XML Proxies is the need to gain 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. XML Proxies 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.

VI. Barriers to Adoption of XML Proxies

Despite all the promises of XML Proxies, they aren't a panacea. In particular, a number of barriers exist to implementing XML Proxies as a way of gaining the benefits described earlier:

- XML and Web Services standards and markets are in a state of flux
- Increased competition from alternate solutions
- Too many network appliances on the "rack"
- Processing overhead introduced by intermediaries.

6.1 XML and Web Services Standards and Markets in Flux

XML and Web Services security, management, transaction, and reliability standards are currently in various different states of completion and adoption.

One of the challenges to implementing XML Proxies is that their underlying standards are rapidly changing. In particular, the security, management, transaction, and reliability standards are currently in various different states of completion. More importantly, they are in different stages of adoption. Some are gaining increased adoption, while some specifications (notably the transaction and reliability standards) have yet to gain much of a foothold.

In addition, the software vendor community has yet to decide upon some significant details with regards to how XML and Web Services calls will flow through intermediaries. For example, it has yet to be decided how to pass information to XML-aware intermediaries, and to identify which Proxies should process which pieces of information. Also, it is not clear if there is a standard mechanism for intermediaries to forward or reroute the information they process. In addition, there is limited adoption of the SOAP Intermediary functionality as described previously in this report. Furthermore, the usage of intermediaries complicates error handling and return messages. As such, it will be more difficult to implement XML Proxies to provide these capabilities in the short term.

However, companies looking to implement XML Proxies should be looking not just at solving immediate challenges, but also at the longer-term benefits of XML Proxies. Many of today's specifications will become accepted standards over the next 12-24 months, and so the XML Proxies that are based on those standards will deliver a greater value. However, today's products can provide value to companies right away. Companies implementing XML Proxy solutions today should expect, however, that they will be operating at the "bleeding edge." As a result, change will be more the rule than the exception. With increased standardization will come increased need to standardize internal XML processing to comply with these standards and realize the benefits described in this report.

Furthermore, the W3C and other standards bodies have recommended that active intermediaries describe the alterations made to SOAP messages in a manner that allows the affected SOAP nodes to detect such modifications. For example, an active intermediary may describe the processing performed by inserting header blocks into the outbound SOAP message that inform downstream SOAP nodes of what operations it performed on the message (such as encryption, signature, and transformation).

6.2 Increased Competition through Product "Scope Creep"

Another barrier to adoption of XML Proxies is the increasing "scope creep"--or expansion of product features and functionality--of different XML and Web Services solutions. As various different solutions add security, management, performance enhancement, transformation, and routing features to their product line, they may end up competing with dedicated XML Proxy solutions. This product/market confusion will lead to inevitable apples-to-oranges comparisons of various different solutions for similar problems. Of course, the key to escaping this market problem is convergence of XML Proxies on a standard set of functions. As users understand that XML Proxies will provide a well-defined set of features and functions, they will be able to decide whether they want the set of features offered by XML Proxies or "best of breed" solutions offered by competing solution categories.

6.3 The Rack "Stack"

In some instances network rack space is more precious than Manhattan real estate.

Some people have expressed the opinion that rack space is in some instances more precious than Manhattan real estate. While this may be debatable at different sized enterprises, there is nevertheless the fact that hardware XML solutions face the additional challenge of making sure their solutions have a place in existing IT administration environments. The glut of Network Appliances and devices over the past five years has resulted in a complex "stack" of rack-mountable appliances covering everything from Voice-over-IP applications to search appliances. Even if there is physically room in the stack, there simply may be no interest of IT administrators to add yet another device to the stack. They may want to see a consolidation of devices rather than addition of new, untested applications.

Clearly, there is some clearing out of rack space going on currently, with a movement to more space and resource-efficient server "blade" configurations. As this process progresses, there will be more room for hardware XML Proxy solutions, but in some cases, the lack of rack space can pose challenges.

6.4 Processing Overhead

The major penalty paid for the functionality of XML Proxies is the necessary overhead placed on the exchange of XML messages. Historically, software firewalls have had significant processor and memory requirements in order to support many simultaneous users. The enterprises add value-added features to XML messages, the more they will notice the delay and latency of the messaging. This penalty is especially the case when the XML Proxy accepts raw XML documents, and the device is responsible for adding security features while transforming the document, performing rule-based routing, and adding management operations. The more the XML Proxy performs parsing, encryption, rule-lookup, and transformation operations, the more latency will be introduced into the network. This latency may be acceptable in asynchronous messaging scenarios (such as the email analogy described earlier), but may not be acceptable in high volume throughput environments.

Hardware XML Network Appliance solutions are the best suited to solving the latency issues introduced by intermediating XML messages. Network appliances utilize specialized hardware and chips to accelerate encryption, parsing, routing, and transformation steps. In this manner, they can reduce the time it takes to perform basic XML operations. However, it is yet to be shown how these devices will perform in high transaction environments.

VII. Future Trends

As the XML Proxy market develops, some key trends will emerge that will help clarify the offerings of XML Proxy vendors:

- The convergence of XML Proxy features on an accepted set of features
- Rapid growth of XML traffic on the Network
- The entrance of established networking vendors into the XML Proxy market
- Further clarification of the role of SOAP Intermediaries.

Each of these trends will help to establish the role of XML Proxies in the IT environment.

7.1 Convergence on a set of functionality: the one-stop box

As enterprises use XML Proxies as XML-aware intermediaries to monitor, manage, and modify XML and Web Services traffic on the network, there will be a demand for these solutions to handle an increasing set of functionality. At some point in the future, it won't be feasible to implement separate solutions for corporate-wide XML security, management, routing, transformation, and performance enhancement. The various different capabilities of XML Proxies identified in this report will be aggregated into a single accepted set of functionality that sets XML Proxy solutions apart from other intermediary approaches. After all, if the device is handling all XML traffic on the network, adding an increased set of value-added features is a natural step. It will initially be easier for software-based XML Proxies to add this functionality first, but as this functionality becomes more established, it is expected that hardware XML Network Appliance solutions will soon follow with similar capabilities.

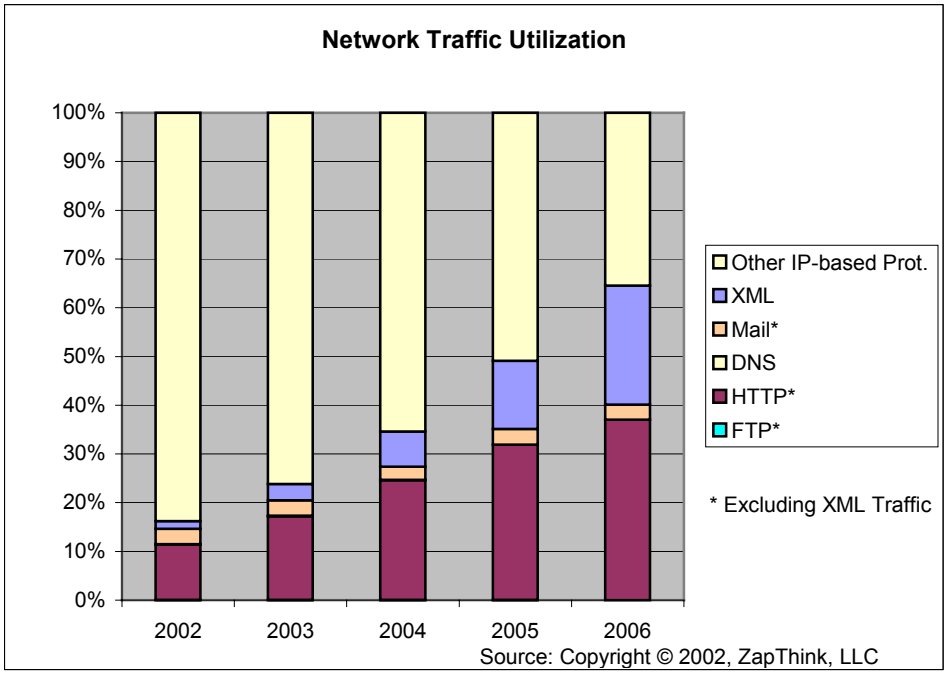
Decision Point

Currently, ZapThink estimates that XML represents only 2% of all network traffic on the network in 2002, however, this is expected to increase to just under 25% of all LAN network traffic by 2006.

7.2 Rapid growth of XML traffic on the network

ZapThink expects XML traffic on the network to greatly increase over the next few years. Currently, ZapThink estimates that XML represents only 2% of all network traffic on the network in 2002, however, this is expected to increase to just under 25% of all LAN network traffic by 2006. The following chart and diagram show the expansion of network traffic over the next few years.

Figure 7.1: Growth of XML as Percentage of Network Traffic†



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

Table 7.1: 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 © 2002 ZapThink, LLC

7.3 Entrance of the "Established" Network Appliance Vendors

As XML Proxy solutions become more visible in the corporate IT environment, it will be a natural play for the current "established" Network Appliance vendor offerings from companies such as **Cisco, Nortel, Intel, F5, CacheFlow,** and **ArrowPoint** to make entrances in this market. Since these vendors are currently experiencing a difficult economic climate that is expected to last into 2003, they will not be apt to take unnecessary risks to enter new markets. However, as the market entrants mentioned in this report validate the market for XML-aware intermediaries, these established vendors will seek to enter and capture this emerging market. Once they enter the space, we can expect to see increasing commoditization of XML Proxy solutions, heightened M&A activity, and competitive pressure.

7.4 Further clarification of the role of SOAP Intermediaries

Another major trend over the next 12-24 months is the clarification of the role of XML Proxies and other SOAP Intermediaries. Currently, intermediaries are somewhat of an overlooked feature of SOAP. However, as vendors and enterprises define and adopt security, transaction, reliability, and management standards, standards groups will need to pay attention to the role that XML-aware intermediaries will play to facilitate these standards. As the standards bodies' attention focuses on this emerging solution category, they will clarify the role of SOAP intermediaries, thus lending additional support and credibility to the XML Proxy market.

VIII. Conclusions

In this report, ZapThink has defines a new category of solution capable of processing and value-adding XML traffic on the network. These XML Proxies, implemented as hardware XML Network Appliances or software XML Proxies, will become increasingly important as users realize that TCP/IP-based firewalls are incapable of adequately handling content-centric data formats. Serving as the central aggregation point for XML traffic on the network, XML Proxies will have to handle a growing set of functionality and capabilities including security, routing, performance enhancement, transformation, and management. XML Proxies must be able to handle internal XML intermediation functions as well as external interface requirements, and do so in a transparent manner.

8.1 Key Notes

- *Increasingly organizations are seeking to find solutions that can transparently monitor XML traffic on the network and apply business rules or corporate IT policies.*
- *XML is creating new exposures–security, performance, and otherwise–that corporate IT environments must appropriately deal with.*
- *In order to process high volumes of XML content on the network, hardware and software devices need to be able to understand not only network protocols, but also the XML-based content traveling on these protocols*
- *Most network device terms have lost their specific meaning when used beyond the third layer of the network stack*
- *The TCP/IP port model is too simplistic for dealing with XML-based content and Web Services traffic in particular.*
- *In many ways, this term "XML Application Firewall" is inaccurate in describing the functionality of XML-aware intermediaries.*
- *The main benefit of implementing XML Proxies in an explicit manner is that users can control when and how XML traffic is processed, without needing to involve central IT administrators.*
- *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.*
- *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.*
- *XML Proxies can implement the latest XML security technologies in a transparent manner.*
- *As XML-aware intermediaries, XML Proxies can be inserted into the XML stream to provide monitoring and management functionality.*
- *XML Proxies can also traverse protocol boundaries--accepting XML requests in one protocol format and spitting it out in another.*

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

- *ZapThink sees an increasing trend towards traditional firewall and network applications "crossing-over" into the XML Proxy market.*
- *XML and Web Services security, management, transaction, and reliability standards are currently in various different states of completion and adoption.*
- *In some instances network rack space is more precious than Manhattan real estate.*

8.2 Decision Points

- *New networking products must transcend the limited OSI model and focus not on the message packet or envelope, but rather the content of the message itself.*
- *XML Proxies are capable of examining traffic at the content level, and can optionally handle other document types such as HTML or EDI content.*
- *The use of SOAP intermediaries can provide an optimal means for inserting XML Proxies into the Web Services exchange process in a transparent manner*
- *There are two primary means to improve the performance of XML on the network: caching and compression.*
- *There are two primary ways that enterprises can implement XML Proxy solutions: in dedicated hardware Network Appliances or in specialized software Proxies.*
- *Centralizing messaging functions in an XML-aware intermediary such as the XML Proxy can greatly improve performance and scalability.*
- *Currently, ZapThink estimates that XML represents only 2% of all network traffic on the network in 2002, however, this is expected to increase to just under 25% of all LAN network traffic by 2006.*

8.3 Figures

- Figure 2.1: OSI Network Model
- Figure 7.1: Growth of XML as Percentage of Network Traffic

8.4 Tables

- Table 7.1: Percentage utilization of Network Traffic by format

IX. Profiled Vendors

9.1 XML-aware Network Appliances

DataPower

Please see ZapNote ZTZN-0132

Forum Systems

Please see ZapNote ZTZN-0212

Sarvega

Please see ZapNote ZTZN-0271

9.2 Software XML Firewalls and Proxies

Actional

Please see ZapNote ZTZN-0280

Primordial

Primordial is no longer in business

Reactivity
Please see ZapNote ZTZN-1082

Vordel
Please see ZapNote ZTZN-0238

Westbridge Technology
Please see ZapNote ZTZN-0612

A. Related Research

Reports

- *Web Services Technologies and Trends* Report (ZTR-WEBSRV)
- *Service-Oriented Integration* Report (ZTR-WS103)
- *XML and Web Services Security* Report (ZTR-WS104)
- *XML Data Store Technologies and Trends* Report (ZTR-ST100)
- *XML in the Content Lifecycle* Report (ZTR-CL100)
- *Service-Oriented Management* Report (ZTR-WS106)
- *Web Services Testing* Report (ZTR-WS105)

B. 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