ARE CURRENT B2B SECURITY STANDARDS ANTITHETICAL TO GLOBAL INTEROPERABILITY?

M.Pather[a] and LM Venter[b]

[a]Faculty of Computer Studies, Port Elizabeth Technikon

[b]Faculty of Computer Science, University of South Africa

[a] maree@petech.ac.za

[b] ventelm@unisa.ac.za

ABSTRACT

This paper argues that powerful impetus exists for the realisation of a global multilateral electronic market infrastructure, as envisaged by the ebXML[1] endeavour. Crucial in this regard is the problem of interoperability: making it possible for disparate systems to interact fluently. An obvious panacea would be to use a pre-specified set of protocols comprised of selected globally-accessible open standards. However, the status quo B2B-interaction implementations comprise protocol-neutral technologies and standards that propagate unwieldy permutations of possible trading partner configurations, which, ultimately, are antithetical to the creation of a global electronic marketplace.

Allowing diversity in the choice of protocols is a limiting factor in cross-integration. It facilitates the relatively smaller numbers of participants to negotiate common terms of agreement for bivalent relationships, but each participant might have to configure a different protocol profile for each trading partner it intends to interact with. Thus, interoperability in the wider global multilateral marketplace context would be stultified, since isolated pockets of discrete relationships are proliferating on an ad-hoc basis. Unless a global initiative in which a set of open standards is unequivocally specified as a single standard for global interoperability, an ebXML-like marketplace will remain a pipedream.

It is clear that Web services provide the most promising means of achieving global interoperability. Yet, within the context of a global multilateral electronic marketplace, unless specific directives are forthcoming at the highest level, standards such as WS-Security[2] remain protocol-neutral and continue to support an opposing momentum.

KEY WORDS

B2B-interaction standards, interoperability, global marketplace

---

[1] http://www.ebxml.org; electronic business extensible markup language

[2] http://www.verisign.com/wss/wss.pdf (contributions from Microsoft, Verisign and IBM)

ARE CURRENT B2B SECURITY STANDARDS ANTITHETICAL TO GLOBAL INTEROPERABILITY?

1 INTRODUCTION

The context of the main argument presented in this paper is a global, multilateral marketplace (as envisaged by the ebXML endeavour). This represents an ideal platform upon which businesses of all types and sizes, from all over the world, can interact by way of a common "language", common semantics, common protocol specifications, common business processes and common business document specifications. The advent of the Web services (Microsoft, 2001) standards has significantly enhanced the possibility of this becoming a reality.

This paper argues that although sufficient motivation exists for the realisation of a global multilateral electronic-market infrastructure current B2B-interactions based on protocol-neutral technologies and standards propagate unwieldy permutations of possible trading partner configurations, which, ultimately, are antithetical to the creation of a global electronic marketplace.

This paper has the following sequence: the nature of B2B-interaction is explored, followed by a brief look at the motivation for a global multilateral marketplace; the key B2B-interaction concepts of integration and interoperability are defined, the problem of multiplicities in respect of possible protocols and configuration nuances is considered, security-specific interoperability is then considered as an example of the problem, followed by a brief look at Web services as a means of counteracting the problem.

2 THE NATURE OF B2B-INTERACTION

Choudury (1997) contends that the business strategy which motivates inter-organisational information systems (equated to B2B-interaction in this discussion) is either *competitive*, e.g., by interacting with an organization in order to woo it away from a rival company; or *cooperative*, in which case it may be either a *strategic alliance* among a few selected organizations, or a *public good* which is open to all organizations. This obviously has implications for how B2B-interaction configurations are selected (in trading partner agreements). These strategies may be observable as three distinct B2B implementation types (as derived from the work of Choudury, 1997 and Kumar and Dissel, 1996):

Multilateral B2B-interaction: This type is depicted by an m:n relationship between buyers and sellers. See Fig. 1 (a) below. It is characterized by a *pooled dependency*, where participants share and use common resources e.g., common databases and common applications, such as is used in airline reservation systems. This type appears to be the obvious "base" type for extrapolation into a common-good, global electronic marketplace.

Electronic Dyads: These are 1:n relationships between collaborators. See Fig. 1 (b) below. Electronic Digital Interchange (EDI) links are a common example. A *reciprocal dependency* exists between organizations in a bilateral fashion. Typically, the organizations may work towards designing, developing and delivering a common product. They usually represent joint ventures, which may be long-term or short-term and may be competitive or cooperative (in a strategic alliance sense).

Electronic Monopolies: This is a special case of electronic dyad - a bilateral relationship, but only one link is established between a buyer and a seller (1:1), as in a single-supplier supply chain.

See Fig.1 (c) below. This is a *sequential dependency*, and represents strategic necessity rather than strategic advantage.

The nature of B2B-interaction has led to a diversity of integration paradigms, typically manifested in B2B-interaction models such as Microsoft's BizTalk (Li, 2002), RosettaNet[3], and ebXML. It is contended in this paper that current models have provided more support for dyadic and monopolistic electronic integration than for a global multilateral electronic marketplace.

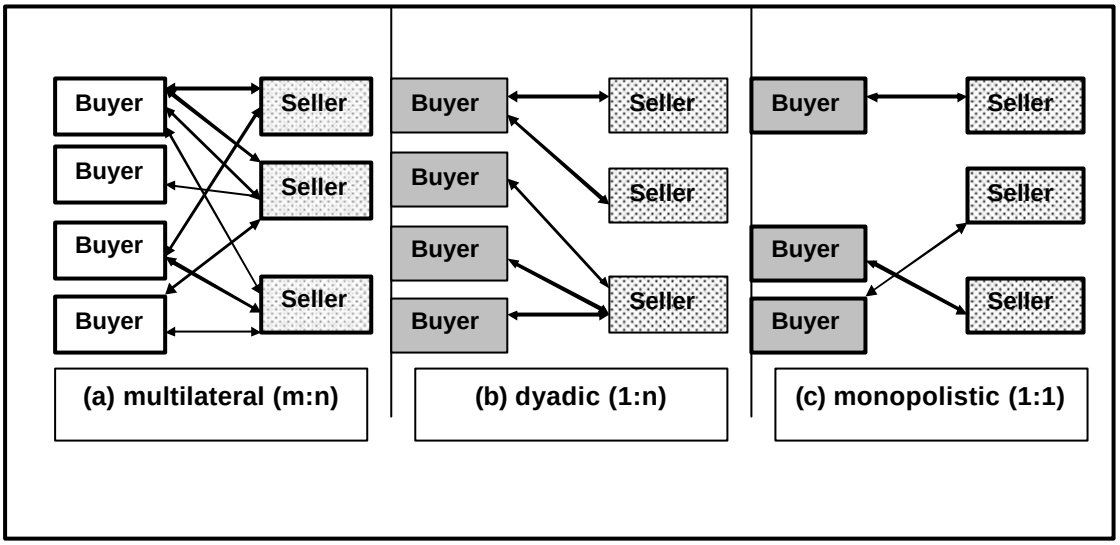

FIG 1: TYPES OF B2B-INTERACTIONS (based on Choudury, 1997)

The next section serves to validate the contemplation a global marketplace infrastructure.

3 HOW REAL IS THE MOTIVATION FOR A GLOBAL MARKETPLACE?

The motivation for supporting a global infrastructure for m:n relationships (based on Kumar and Dissel, 1996, for the implementation of inter-organisational information systems) include:

Globalization: This refers to the trend to expand trading world-wide, the "global village" concept. The Internet is the primary facilitator of this trend. The newer tools for the discovery of trading partners together with new collaboration platforms, as envisaged by the UN-CEFACT-OASIS[4] ebXML global electronic marketplace are intended to buttress this drive.

Environmental turbulence: Business process re-engineering is driven by ever-evolving technologies; companies both want to be competitive and to be seen to be using cutting-edge technologies. It is generally accepted in the literature that the next wave in software development is towards service-oriented architectures (SOA's) and Grid Computing (outsourced computing/ processing resources). More precisely, the shift is towards Services-based Application Service Providers (ASP's) and Computing Service Providers (CSP's) ("The Next Big Thing", Economist, 2004, for example). Forrester Research (cited in Knorr, 2003) indicates that 85% of their Fortune 500 respondents intended to deploy Web services (discussed later) in 2003. Again, the degree of interaction propounded by this trend would increasingly rely on a standardised global marketplace infrastructure.

---

[3] http://www.rosettanet.org
[4] UN-CEFACT: United Nations' Centre for Trade Facilitation and Electronic Business
OASIS: Organisation for the Advancement of Structured Information Standards

Resource pooling: Business partners can share resources in order to lower costs; code-reuse pools (as in utilizing common Web services and related UDDI[5] servers) and ebXML Registries and Repositories would provide avenues for resource pooling. Grid Computing will probably epitomize this strategy, especially if the envisaged pay-per-use implementation comes to pass (Ibid).

Risk-sharing: Business processes and resources can also be shared in order to reduce risk; the developing of standards is typically an interest-group effort with participants originating globally, thereby reducing the risk of failure. Again, major ASP's and CSP's will provide risk-reduction opportunities for individual companies, via comprehensive, selectable outsourced security services. Global risk-sharing resources are the obvious next step.

Reducing supply-chain uncertainty: Product availability and fulfilment procedures have become more effective with online communication. Global m:n relationships based on pre-specified standards would ensure more competitive services in this regard.

Increasing resource utilization: This refers to ensuring more efficient use of equipment and skills. In a Grid Computing scenario, computing resources would be made available, when, and to the extent that they are required, by a third-party vendor.

It may therefore be presumed that sufficient impetus towards many-to-many (m:n) relationships in a multilateral, global marketplace context (notwithstanding the existence of other forms of B2B relationships) certainly does exist. The most significant aspect of this scenario is the integration of systems across the globe. Integration relies on disparate systems being made interoperable. These concepts are discussed in the next section.

4 KEY CONCEPTS IN B2B-INTERACTION: INTEGRATION AND INTEROPERABILITY

It is common knowledge that variations in business practices, and corresponding business documents, are innumerable. Similarly, systems for B2B-interaction may be comprised of innumerable permutations of hardware platforms, operating systems, programming languages, transport protocols, messaging protocols, security protocols, etc. See Fig 2 below. These are factors that contribute to the multiplicity paradox discussed in the next section. B2B-interaction, between both internal and external enterprise applications is generally complicated by issues such as scalability, volatility autonomy, heterogeneity and legacy systems.

In order to accommodate participants, implementers ensure flexibility with regard to these factors, generally configured in an ad hoc manner. As mentioned previously, this generally works for 1:1 and 1:n relationships. Typical B2B platforms such as EDI (electronic data interchange), Microsoft's BizTalk, OASIS-UN-CEFACT's ebXML, and others, feasibly facilitate integration in 1:1 relationships and even 1:n relationships, but in m:n relationships where m and n represent large numbers of disparate systems, such as in a global marketplace, the permutation possibilities escalate exponentially.

Integration is, however, of paramount importance if the global marketplace vision is ever to be realized. Integration here refers to the degree to which organizations are linked by their software and hardware, share common semantics and data, and use commonly-agreed-upon (pre-specified) management structures, business processes and business practices. The building of applications able to operate with each other across a global integration chasm – i.e. addressing *global interoperability* - is the challenge of B2B-integration in a global multilateral marketplace context.

[5] Universal Description and Discovery and Integration; http://uddi.microsoft.com

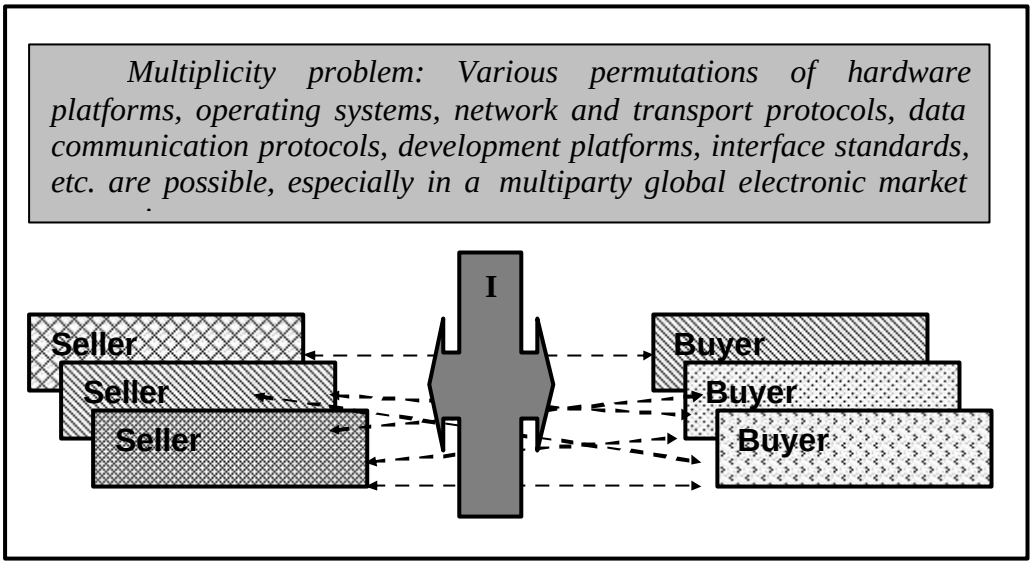

Fig. 2: The integration problem in multilateral B2B-interaction

The International Standards Organisation Technical Committee (as cited in Rayman, 1999) defines *interoperability* as:

"The ability of systems to provide services to and accept services from other systems and to use the services exchanged to enable them to operate effectively together"

Interoperability thus involves exchanging services which may originate from disparate systems, but have the ability to operate effectively together (inter-operate). In a global multilateral B2B-interaction scenario, finding the means to ensure sufficient interoperability between all collaborators is vital. For interoperability, systems do not have to be homogeneous; it is not necessary for the applications to possess the same design or internal architecture.

What is critical, however, is that applications supplying services (providers) implement a set of specifications which makes the services accessible by *all* the applications requesting the services (consumers). In the global marketplace scenario, these specifications must be universally usable and universally accepted as standard.

5 THE MULTIPLICITY PARADOX

In terms of the B2B types described above, it is contended here that many current models for B2B-interaction have provided more support for dyadic and monopolistic electronic integration than for a global multilateral electronic marketplace. For instance, allowing for a wide range of protocols serves to *accommodate* a greater number of configuration nuances prevailing in disparate systems in 1:1 and 1:n relationships. Examples of this are the ebXML's Collaboration Protocol Profile and RosettaNet's Implementation Framework (RNIF) which may be pre-specified by each collaboration partner. While this promotes integration between pairs of collaborating systems, a contrary effect is brought to bear on global uniformity.

ebXML is described as a suite of specifications for providing interoperability in a global marketplace. However, it makes provision for trading partner agreements (Collaboration Profile Agreements) which fosters discrete 1: n and 1:1 collaboration pockets. Trading partners may specify, inter alia, security, messaging and transport protocols in a Collaboration Partner Profile.

Even company-specific business process specifications may be submitted to the ebXML Registry/ Repository. It is contended here that, in terms of establishing a global interconnected marketplace, this approach can be considered *laissez faire*. What is required is a prescribed set of specifications for m:n relationships, which should not necessarily preclude 1:1 and 1:n arrangements. This certainly promotes interoperability in these circumstances, but allowing multiplicity in protocol choice is a limiting factor for cross-integration. It facilitates the relatively smaller numbers of participants to negotiate common terms of agreement for bivalent relationships, but each participant might have to configure a different protocol profile for each trading partner it intends to interact with. As the number of trading partners increases, so the number of possible combinations of protocol profiles increases exponentially. Thus, interoperability in a global multilateral marketplace context would be stultified, since isolated pockets of discrete relationships would proliferate on an ad-hoc basis.

In a global marketplace infrastructure these multiplicities would have to be reconciled, to facilitate m:n cross-integration. Therefore, unless a global initiative in which a set of open standards is unequivocally specified as a single standard for global interoperability, an ebXML-like marketplace will remain a pipedream.

The obvious solution to this multiplicity problem would seem to rest on a global m:n standards-infrastructure, wherein a set of protocols is unequivocally specified (and managed). Ideally, a central Registry/Repository system (similar to the Public Key Infrastructure/PKI hierarchical arrangement) would provide inheritable objects and common semantics (including programming interfaces and classes) for Web service components (common core components), common document templates (XML schema documents), and protocol specifications for security, messaging and transport.

6 SECURITY INTEROPERABILITY

One of the elements requiring pre-specification between trading partners is security. It has become common practice to use tools such as network firewalls and content filters to protect the internal network from outside intruders. However, in a global B2B environment, the following have to be secured: the internal network, the Web server, the application server, the database, the application and the messaging system between application interfaces. Each of these can be implemented and configured in a variety of ways:

- What authentication mechanism should be used? Should it occur at the operating system level (user identity + security context) or at the application level (generally, user identity only)?
- Where should authorization take place? Should this occur at the resource level (Web Server resources: Web pages, or Web Services; Database Resources; Network Resources: file system, shared folders, Directory Services, etc; System resources: Registry, Event Logs, configuration files, etc.) or at the application level (role-based; typically at the method level of components).
- How will the user's identity be persisted through the various layers/tiers of the application? Will the identity flow from the Web Server to the Application Files to the Database? If so, in what manner? (Meier et al, 2004).

What are the implications for heterogeneous systems? The possible permutations for security configurations in m:n interactions are quite formidable. Considering the anticipated gravitation towards SOA's, which are generally Web services-based, it is pertinent to examine how the Web Services Security standard (WS-Security)[6] addresses interoperability in the wider context of global

cooperation. As with ebXML, *WS-Security contributes to the multiplicity paradox* by not specifying explicit protocols to be used (underlining added for emphasis):

"This specification is intended to provide a flexible set of mechanisms that can be used to construct a range of security protocols; in other words this specification intentionally does not describe explicit fixed security protocols ". (Goals and Requirements, Lines 133-135).

Thus, the WS-Security standard, although targeting integration and interoperability, by being flexible (protocol-neutral), actually promotes esoteric configurations in a global context.

7 ADDRESSING THE INTEROPERABILITY PROBLEM

7.1 An interface to ensure interoperability

The concept of an "*interface*" is central to interoperability, but the technical specifications of interfaces need to be *open*, i.e. "readily and non-discriminatorily available to all vendors, service providers, and users, and if such specifications are revised only with timely notice and public process" (Eurobit-ITIC-JEIDA Paper, January 1995, cited in Williams et al, 2000). Interfaces may roughly be considered as translators and are represented by the shaded area indicated by I in Fig. 2 above. Open-interface technologies are essential to solving the global marketplace interoperability problem. However, if *n* (large number of) applications are to be configured to interact via standard interfaces (with common interoperability characteristics), the interface characteristics must be *pre-specified at the required authority level* (such as the World Wide Web Consortium, W3C).

In addition to the interface specifications, a common "language" usable by all applications (via a common interface) is also a prerequisite. This language is already widely accepted to be XML (for data description, data transformation, SOAP messaging, and Web services discovery and description), which can be processed in newer Internet Browser software and database management systems. To standardise a global XML grammar for global use (a lingua franca), a common vocabulary (mainly for business terms, programming identifiers and function parameters) and common semantics (meanings of generic words used) would be necessary. Any particular pre-specified business process must occur in a pre-specified business workflow, specified using the common semantics and the common vocabulary; it must also comprise business documents using pre-specified XML specification documents (XSD's). Application interfaces *must* be created according to one (W3C) comprehensive, *pre-specified* interoperability standard comprising open standards, such as:

- data description language (e.g. XML)
- network protocols, e.g. Internet Protocol (IP),
- transport protocols, e.g. Hypertext Transfer Protocol (HTTP) over Transmission Control Protocol (TCP),
- messaging protocols, e.g. Simple Object Access Protocol (SOAP[7]) with Multimedia Internet Mail Extensions (MIME) attachments and
- security protocols, e.g. Transport Layer Security (TLS) for internal networks, Secure Internet Protocol (IPSec) for SOAP messaging between trading partners, XML encryption for encrypting parts of the XML attachment at the application level, Advanced Encryption Standard (AES) for encryption, XML digital signatures for integrity control at the messaging level, Security Assertion Markup Language (SAML) / XML Access Control Markup Language (XACML) for Web service access control, etc.

7 http://www.w3.org/TR/SOAP

A Web service represents a convenient standard application interface in this regard, allowing disparate systems to communicate/integrate. A Web service may be briefly described as programmable application logic accessible via standard Web protocols (Wall L and Lader A, 2002: 220-221). The W3C's Web Service Architecture Group defines a Web service as follows: "A Web service is a software application identified by a URI, whose interfaces and bindings are capable of being defined, described and discovered by XML objects and supports direct interactions with other software applications using XML-based messages via Internet-based protocols" (Jenz, 2002).

The major benefits of Web services as an application interface may be listed as follows:

- Web services can be used either within an organization or can be made available over the Internet for use by other organizations.
- Web services are accessible through a standard interface which allows disparate systems to interact.
- As Web services utilize XML-based messaging, they promote interoperability on both data and system levels. As XML is (UNICODE) text-based, it can be processed by any computing platform.
- Web services allow a high degree of abstraction between the implementation and consumption of a service. Therefore, both the provider and consumer systems require only knowing the inputs, outputs and location of the Web service.
- In its simplest form, a Web service may be accessed from any system type as follows: the uniform resource identifier (URI) of the Web service is entered on a Web browser; the browser automatically generates a local client through which the service can be activated; input is entered via the client and the output(s) of the service is returned to the browser client. A more comprehensive client for handling the output (and automated entering of input) can be used in place of the browser.
- Messaging between the client and the Web service is by way of Simple Object Access Protocol (SOAP), which is an XML-based open standard for messaging.

7.2 Web Services infrastructure

Web services use an infrastructure that provides for the discovery and publishing of services over the Internet. This infrastructure consists of Web service directories, a Web service discovery mechanism, a Web service description mechanism and open-standard wire formats.

The Universal Description, Discovery and Integration (UDDI) standard has been designed for implementing directories (registry and query facilities) for Web Services. Additionally, the Web Services Description Language (WSDL) standard supports a Web service description and discovery mechanism. The deployment of Web Services is provided by SOAP. As all of these open standards are based on XML, the entire Web services family of standards provides a suitable platform for B2B integration.

Web services directories (UDDI registries) provide a central location where Web services can be published and discovered. Currently Web services may be published ad lib, which also does not support a global interoperability scenario.

The *Web service discovery* mechanism ensures that one or more related documents that describe and contain links to a particular Web service can be located. This is achieved through the use of WSDL which defines an XML grammar for describing communications in a structured way. Through this mechanism the existence of Web services and their location is learned.

The *service description* is an XML document written in WSDL that defines the format of messages the Web service understands. The service description serves as an agreement that defines the behaviour of a Web service and provides instructions for how potential clients can interact with

it. The behaviour of a Web service is determined by messaging patterns that the service defines and supports. These patterns theoretically dictate what the service consumer can expect to happen when a properly formatted message is submitted to the Web service.

The *service wire formats* provide universal communication through open protocols. The key protocol for Web service communication is SOAP. The SOAP specification specifies all the rules for locating Web services, integrating them into applications and communicating between them. It does not define any application or transport semantics and this allows it to be used in a variety of systems. (Microsoft, 2001). Again, this adds to the complexity of current B2B interaction, even where Web services are used for interoperability

8 CONCLUSION

It therefore seems quite likely that m:n B2B-interactions will have to co-exist with 1:n and 1:1 B2B- interactions indefinitely. However, just as the Internet has supplanted most private networks with Intranets and Extranets, so too a global marketplace infrastructure can augment (if not supplant) current Extranet/VPN networks. There is certainly sufficient incentive for enterprises (even non-business organisations, such as Interpol and Research institutions) that may wish to collaborate seamlessly on an m:n basis (even in a cross-industry fashion). The same physical resources (Internet infrastructure, computer/communication hardware, etc) can be used; packet-based messaging and multiplexing of signals allow for a great deal of simultaneous but dissimilar functionality (the various configuration nuances for bivalent interactions are primarily incorporated within the end systems). For participating in the m:n marketplace, each bivalent interaction will be configured in the same way (protocols and settings, possibly even documents and business processes). For non-m:n marketplace interactions, different collaboration agreements might apply. It seems quite likely that the latter could gravitate towards the more common m:n specification suite to obviate redundancy, especially in cases where a company requires n:m and 1:n or 1:1 interaction systems.

The motivation for Web services is primarily to provide interoperability among disparate systems. It seems conceivable that either a superset (even a sub-set) of the same standards can be improvised to allow for a single global standard specification suite, to create the context for an m:n global marketplace. Covisint[8] is an attempt by the automotive industry to create an industry-specific m:n marketplace for diverse participants. ebXML endeavoured to link businesses of various types, size and location, but failed to ensure interoperability by falling prey to the multiplicity paradox discussed earlier.

It is clear that Web services provide the most promising basis for an interoperability platform. However, within the context of a global multilateral electronic marketplace, unless specific directives are forthcoming at the highest level, current protocol-neutral standards and ad-hoc configurations will propagate an antithetical situation.

9 REFERENCES

Choudury, V. (1997). Strategic Choices in the Development of Inter-organizational Information Systems. *Information Systems Research 8(1): 1-24.*

Economist (2004). “The next big thing”. Editorial.

Jenz, D. E. (2002). ebXML and Web Services - Friends or Foes? Retrieved May 12, 2003, from http://www.webservices.org/index.php/article/articleview/451/1/22/

Kumar, K. and Dissel, H.G. (1996). Sustainable Collaboration : Managing Conflict and Cooperation in Interorganizational Systems. *MIS Quarterly September: 279-299*

Knorr, E. (2003). “BluePrint for Web Services”. InfoWorld.com (12.01.03).

[8] http://www.covisint.com

Meier, J.D., Mackman, A., Vasireddy, S., Dunner, M., Escamilla, R. and Murukan, A. (2004). Improving Web Application Security. Threats and Countermeasures. Microsoft Press.

Microsoft. (2001). XML Web Services Overview. Retrieved May 29, 2003, from

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpcon Webservicesoverview.asp

Li, P. (2002). *BizTalk Server Developer's guide.* California: McGraw-Hill/Osborne.

Rayman, C. (1999). *NTCIP Introduction/ NTCIP: Infrastructure Interoperability.* Retrieved July 11, 2003, from http://www.itscanada.ca/html/AGM1999/present/ntcip/

Wall L and Lader A (2002). Building Web Services and .NET Applications. McGraw-Hill/Osborne.

Williams, H., Whalley, J., & Li, F. (2000). Interoperability and Electronic Commerce: A New Policy Framework for Evaluating Strategic Options. *Journal of Computer-Mediated Communication 5, (3).* Retrieved June 2, 2003, from http://www.ascusc.org/ jcmc/vol5/issue3/ williams.html

Web Services Security (WS-Security) Version 1 (2002). Retrieved December 2003 from www.verisign.com/wss/wss.pdf.

The financial assistance of National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the authors and are not necessarily to be attributed to the National Research Foundation.