Towards a Negotiation Protocol for ebXML

Hércules Antonio do Prado[1,2], Edilson Ferneda[1], Gentil José de Lucena Filho[3], Aluizio Haendchen Filho[4] and Newton Castilho Lavoyer[1]

[1]*Graduate Program on Knowledge and IT Management, Catholic University of Brasilia, SGAN 916 Av. W5, 70790-160, Brasília, DF, Brazil*

[2]*Secretariat for Management and Strategy, Embrapa, Parque Estação Biológica, PqEB s/n°, 70770-90, Brasília, DF, Brazil*

[3]*Laboratory for Research on Conversations in Organizations, Homero Reis and Consultants, SCN - Q. 01 - Bld. E-50- Room 1102 Ed. Central Park, CEP 70.711-903, Brasília, DF, Brazil*

[4]*UNIDAVI - Universidade para o Desenvolvimento do Alto Vale do Itajaí, Rua Dr. Guilherme Guemballa, Rio do Sul, SC, Brazil*

Keywords: Electronic Business, Web Services, e-Business, B2B, Ontology of Language.

Abstract: In the last decades many research efforts have been devoted to improve electronic business among partner enterprises. The well succeeded results of these efforts led to the widespread use of tools like eCO-Framework (Macro-Economic Framework), EDI (Electronic Document Interchange) and SWIFT (Society for Worldwide Interbank Financial Telecommunication). More recently, the ebXML standard has been developed to expand the B2B (Business to Business) practice, assuring security at a low cost and enabling the commerce among small and medium businesses. However, ebXML provides only physical connection among the parts, lacking support for a business protocol capable of helping a negotiation. A well succeeded negotiation requires a dialog, involving customers and suppliers, product specification, requests, offers, requirements, all in a cycle of successive refinement of expectations. This paper aims at contributing to bridge this gap by means of a conceptual model for negotiation based on the client satisfaction cycle of Flores. Additionally, we demonstrate the electronic commerce route among the enterprises, describe some patterns that led to the arising of ebXML and that are part of its current core, specify the ebXML and propose some trends for the B2B commerce.

1 INTRODUCTION

The electronic business among enterprises, aka B2B, appeared in the 1960s, along with the first systems for Electronic Documents Interchange (EDI). The advances of communication technology and the Internet, along with the development of the eXtensible Markup Language (XML) enabled a more secure and cheap business practice. These changes took the advantages of electronic commerce to small and medium business, beyond the big ones, leading to a joint effort from the Organization for Advancement of Structured Information Standards (OASIS) and United Nations Center for Trade Facilitation and Electronic Business (UN/CEFACT). This led, in 1999, to the development of the specification Electronic Business XML (ebXML).

The paper is organized as follows. In Section 2, a historic context for its proposal is provided by means of a short description of other related standards. The XML specifications, Web Services Description Language (WSDL), Universal Description, Discovery and Integration (UDDI), Simple Object Access Protocol (SOAP), and EDI are discussed. Section 3 presents ebXML standard and, in Section 4, the business process encompassed by ebXML is discussed. The connection between ebXML and the Client Satisfaction Cycle (Flores, 1997) is clarified in Section 5, along with a discussion on how the conversations requirements in business could be covered by the proposed model. A logical structure is presented that encompasses the elements involved in this proposal. Finally, the ongoing work is described in Section 6.

do Prado H., Ferneda E., de Lucena Filho G., Haendchen Filho A. and Castilho Lavoyer N..
Towards a Negotiation Protocol for ebXML.
DOI: 10.5220/0004098602770282
In *Proceedings of the 14th International Conference on Enterprise Information Systems* (ICEIS-2012), pages 277-282
ISBN: 978-989-8565-11-2

2 ebXML PREDECESSORS

2.1 XML

XML standard was developed by World Wide Web Consortium (W3C) in order to address the necessity of interoperability among diverse applications. It was based on Standard Generalized Markup Language (SGML), a standard used for the electronic publishing industry that provides support for text mark-up.

XML, like HTML, is called a mark-up language because it allows structuring a text by means of marking tags. Text characteristics may be indicated by posing a tag marking in the beginning of the text fragment and another tag marking at the end of the fragment.

The W3C objective, as described in the XML specification (W3C, 2004a), was to define a portable language for data description, by means of a new mark-up model, useful to write Web documents. XML provides the author features for defining and creating its own marking labels and attributes, instead of restricting the mark-up schema of HTML, which adopts fixed tags.

The most important characteristic of XML is the ability to interchange data. It allows, for example, that an enterprise, or even different departments from the same enterprise, to interchange XML-marked data through the Web, independently on how the source system is organized. In other words, each part involved in data interchange can use different tools to generate and treat XML documents (Tidwell, 2002).

2.2 Web Services

Web services are applications that, once published, can be found and referenced by HTTP and XML standards like SOAP, UDDI and WSDL (Smith and Vinoski, 2001).

The web services architectures, as specified by W3C (2004b), presents the following components: *Registry* is the place where the services are published, being available for search. The information is organized according the UDDI standard; the consumer of a given service or *client* is any entity that uses the actual or even another service; *Service Provider* is the entity that publishes the service, making it available for search. In order to encapsulate information, the service publishing applies the SOAP protocol, mapping such information in UDDI registry.

2.3 EDI

EDI is a standard that enable the interchange of business documents among commercial partners (Graham, 2003). Likewise, it can be defined as the interchange of documents in a standardized and automatized way among organizations (Hakvoort, 2004). According to Rawlins (2002), EDI accomplishes the following functions (See Figure 1): Format conversion from EDI to the application format; Keeping track of these conversions; Documents interchange among business partners; Generation and acknowledgments follow-up; Auditing and logging; and Negotiation management among the parts.

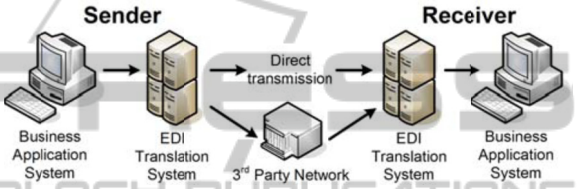

Figure 1: Information flow in EDI (Hakvoort, 2004).

The client application transmits its business data to be formatted by the EDI translator, which in turn converts these data into a message in EDI format. This can be sent directly or by means of a private network to the EDI receptor, which performs a new translation that shall be understandable by the addressed application.

The enterprises have used EDI to interchange business documents, but the lack of standardization, the high cost of implementation and the use of private networks have strongly restricted its application.

3 THE ebXML STANDARD AND ITS BUSINESS PROCESS

When OASIS and UN/CEFACT joined efforts to specify a new XML-based standard that would enable the electronic commerce in Internet, the first one became responsible for defining the technical infrastructure and the XML issues. On the other hand, UN/CEFACT assumed the responsibility for the technical specification of the core components (Hakvoort, 2004).

The first release of ebXML specifications was published in 2001 and, in 2004 ISO approved ebXML as the 15000 standard, according the following set of specifications: ISO 15000-1 ebXML

Collaborative Partner Profile Agreement (ebCPP); ISO 15000-2 ebXML Messaging Service (ebMS); ISO 15000-3 ebXML Registry Information Model (ebRIN); ISO 15000-4 ebXML Registry Services (ebRS). In 2005 the specification 15000-5 ebXML Core Components (ebCC) was also approved.

According to Liang (2012), the ebXML standard was proposed to "provide an open, XML-based infrastructure to enable the global use of e-business information in an interoperable, secure, and consistent manner by all trading partners, targeting to the B2B market." The standard has enabled the B2B practice among small and medium enterprises.

3.1 ebXML Components

The practice of electronic cooperation among commercial partners requires the definition of a business process. In order to achieve a common objective (Irani, 2002), a business process defines rules and responsibilities for each partner and is composed by a set of activities that are performed in a pipeline.

The architectural model of ebXML defines two kinds of visions that cover the relevant aspects of business interactions. These visions were inherited from OpenEDI of UN/CEFACT and define operational and functional issues for performing an electronic cooperation among business partners. The operational vision of a business for Business Operational View (BOV) includes conventions, agreements, mutual obligations, and requirements for the commercial integration. The Functional Service View (FSV) addresses the support services for establishing a negotiation under ebXML and has three phases: implementation, discovery and publishing, and execution. The first phase describes the procedures for creating an ebXML application. Discovery and publishing refers to the search and discovery of services in the ebXML registry. The third phase addresses the scenarios for executing the ebXML transactions (Webber, Dutton, 2000).

The ebXML standard defines an architecture in modules that offer a variety of components that assure the interoperability among the business processes regarding not only the technical infrastructure but also commercial aspects of the business (ebXML, 2012). The architecture encompasses the following components: Core Components (ebCC) – libraries of reusable and extensible components able to support the business processes; Business Process (ebBP) – composed by metamodels of the enterprise business processes; Collaboration Protocol Profile (CPP) and Collaboration Protocol Agreement (CPA) – allows an enterprise to specify its business process and establish the collaboration rules with its partners; Registry (ebRS): repository for business objects; and e-business Message Service (ebMS): provides the message interchange structure that enables interoperability among the business partners. Figure 2 depicts the technical infrastructure of ebXML.

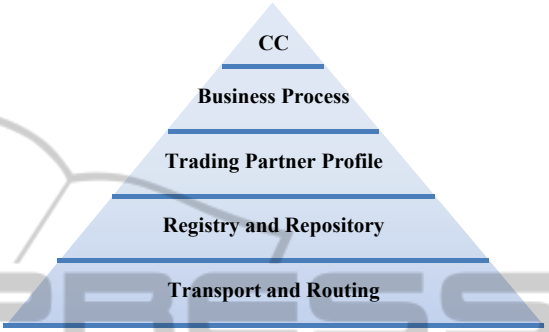

Figure 2: ebXML components.

3.2 Core Components

According to the specification document (ebXML, 2012), the Core Components (ebCC) are defined in such a way that it can be shared by many business types. They persist in the Registry and can be accessed as models for defining a business process for an organization.

A component represents a business concept from the real world and holds a set of data that encompasses a minimum set of business information. An analogy that clarifies this concept is the building blocks in which the blocks can be combined for more complex components. The core components are written in XML and are stored and retrieved from a core library in the ebXML registry. The core components are designed to be reused and extended to define any business process. Additionally, an enterprise developing an ebXML application can define new components that can be published in the registry and become part of the core component's library (ebXML, 2012).

3.3 Business Process

The Business Process (ebBP) models describe the operation of the enterprise business process by means of scenario specifications that must be clear enough to be understandable by other enterprises. This allows the integration of business processes among partners. The models are based on ebXML metamodels and are designed in UML and UMM.

3.4 CPP and CPA

CPP (ebXML, 2012) is presented as a XML document, according to a Document Type Definition (DTD) kept in ebRS. It specifies the technical details and information related to the capability of an enterprise to support a given business process in an electronic collaboration. CPP represents information related to: the location and the enterprise contacts, industry classification, messages and transportation protocol applied by the enterprise, security restrictions, and interfaces used to enable interchange of business documents.

CPP is related to one or more business process supported by a business partner, as well as to the rules accepted by this partner for each business process. An enterprise interested in taking part of a commercial collaboration process using ebXML should generate the CPP document, describing its business profile and publishing it in ebRS, allowing potential business partners to understand the process. The rules in CPP are defined in the Business Process Specification (BPS).

CPA is an XML document, also defined by means of a DTD in ebRS, and specifies the information interchange among business partners, in an ebXML collaboration process. CPA represents the intersection of the CPP business partners involved in a commercial interaction and keeps the results of the partners' agreement. The rules to be followed by the partners must be described in CPA, and include transportation protocols and business messages, security resources (eg, encryption and authentication), and the definition of the business process to be performed (ebXML, 2012).

3.5 Registry Services

The Registry Services (ebRS) component is a repository that provides a set of services to enable information sharing. Information is stored as objects in the repository and managed by ebRS specific processes. An item in the repository refers to an object submitted to be stored, for example an XML document representing an invoice. Each item in the repository is described by means of metadata as an ebRS entry (ebXML, 2002).

3.6 Message Services

The Message Services (ebMS) component defines a set of formats and protocols for message interchange and the use of HTTP and SMTP protocols (Hofreiter et al., 2002). It is based on SOAP extensions, including SOAP Message Attachment that enables the attachment of documents and other items to the message (Hakvoort, 2004).

This component is organized in three levels: an interface to access the service, methods to handle the service, and the mapping of the transportation services (HTTP, SMTP, FTP e IIOP). The methods to handle the service provides the functions: Processing Manager, responsible for building the handling based upon invoked application parameters; Security Service, that authenticates and authorizes the signatures; Message Reliability Service, responding for persistence, retransmission, error notification, and message reception; Packaging Service, which handles the message packaging inside the SOAP packages along with the attachments; and Errors Management Service, that catalogs the errors and keeps track of them. Figure 3 depicts ebMS and respective levels.

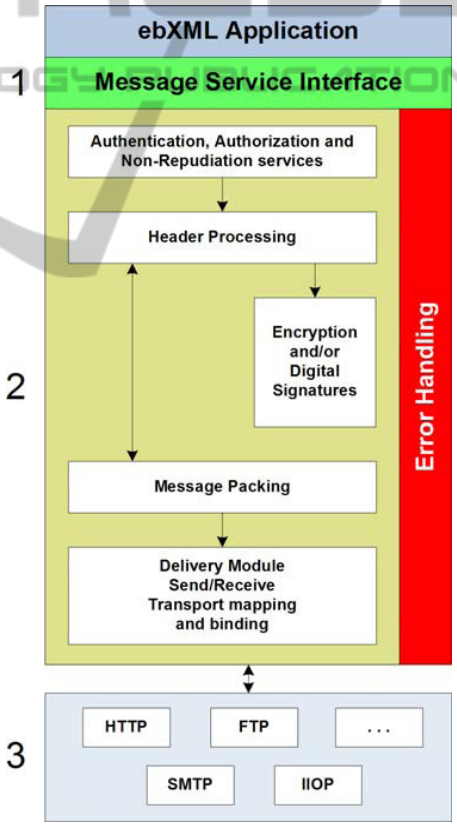

Figure 3: Message services (Hakvoort, 2004).

3.7 ebXML Business Process

The business process in ebXML (Figure 4) starts when a company interested in doing business accesses the internet and query the scenarios and

profiles available on ebXML repository (step 1). After reviewing the business models, it decides to build its own business application, which is made by means of a CPP document.

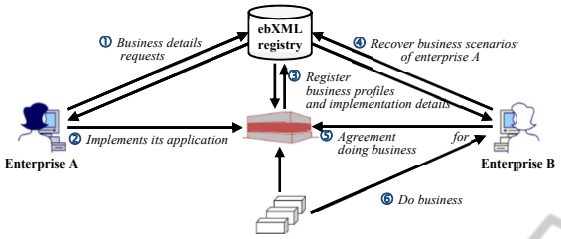

Figure 4: Interaction B2B under ebXML (Hakvoort, 2004).

The document contains all information (eg, technical capabilities, restrictions, and scenarios) that can be interesting for a potential business partner (step 2). Next, enterprise A publishes its CPP document in ebXML (step 3). Enterprise B, which is looking for partners, finds the profile of A and considers it interesting (step 4). Enterprise B requests the CPP document from A and starts the negotiation phase, departing from the CPP or a CPA to build an agreement (step 5). After establishing an agreement, A and B are ready to make electronic commerce in the ebXML context (step 6).

4 THE CLIENT SATISFACTION CYCLE

In their original work, Winograd and Flores (1987) established a challenging vision of model organizations defined as commitments networks. The Client Satisfaction Cycle (CSC) (Flores, 1997) emerged from this concept (see Figure 5).

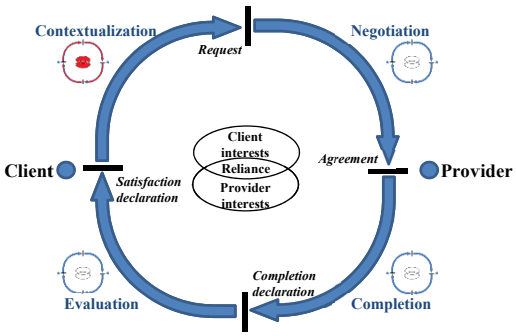

Figure 5: Client satisfaction cycle (Flores, 1997).

CSC provides a model to understand and represent, in a formal and complete way, the generic relations between partners in a business transaction. In this paper, we instantiate CSC aiming at providing more flexibility to the business relations occurring in ebXML. According to Flores (1997), taken as commitments networks, the organizations are constituted or, in other terms, get their *identities* from the conversations that arise and develop inside their environment. Every social organization, whatever it is (eg, a company, public agency, a university, a family, a drugstore, etc.) once constituted, becomes unique and distinguished on the basis of the power that characterize its conversations. These conversations are social phenomena from which work is established and occur. In these conversations, *actions* are developed, *concerns* arise, *assessments* are made and *possibilities* are opened or closed. Linguistically, all these *activities* are manifest in the form of "speech acts" like *requests, promises, declarations* and *statements* which are not mere circumstances of the moment, but rather, are recurrent structures that pervade human interactions. At the same time, these conversations, as human manifestations, besides mere *rational* issues, also carry in its nature *body* and *emotional* components which derive from the human nature of the beings that generate them.

5 THE BUSINESS LINK FOR ebXML

Any negotiation involving an ebXML transaction uses a design protocol departing from inflexible situations given by the available business models. The possibility of a real negotiation, including successive offers and requests usually does not happen, precluding the parts from a richer interaction beyond predefined models. Based on a broad theory of speech acts, Flores (1997) studied the relations existing in negotiations. He proposes a conversation cycle between the client and the provider, defining roles and structuring flows which facilitates the negotiation until its completion. In order to achieve a more flexible negotiation, a framework is proposed based on mappings between the conversation cycle of Flores and the ebXML negotiation cycle. This framework can be considered a logical layer on ebXML platform.

In spite of the wide scope of the dimensions (language, emotion, and body) involved in conversations, in the present work we are only exploring the linguistic dimension. That is, we restrict our considerations to the *speech* and *listening*

acts immersed in the conversations that generate the commitments in business transactions occurring in the ebXML virtual space. An important challenge for this proposal is to understand the types of conversations that occur during an electronic business transaction. Moreover, these conversations must be mapped, classified, and organized. Work is being carried out towards finding and characterizing a conversational ontology able to: capture concepts and relations inside the business domain; derive a set of axioms that enable translating and modeling a B2B representation; and capture subtle issues typical from human nature such as the ones occurring in electronic business transactions. In order to envision the mode and context of these conversations, a mapping from the elements of CSC and the negotiation cycle of ebXML is proposed (see Figure 6). The relationship between the CSC and ebXML are only graphically presented due to space limitations.

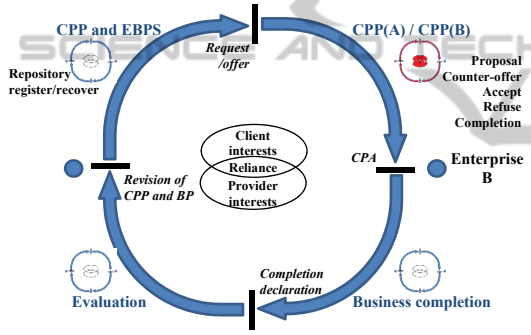

Figure 6: CSC and ebXML components.

6 ONGOING WORK

The current efforts include the creation of an agent organization and the specification of a normative environment in which possible behaviors in the ebXML context can be defined. Agents can be used to establish automatic or semiautomatic conversations or even to help in the negotiations. In this context, some questions like the following must be considered: What are the most likely or possible conversations? Are them recurrent? Can they be classified, organized, mapped? According to Paes et al (2005), the interaction among agents in an organization can be considered in the following levels: content level, related to the content of the information interchanged between agents; intention level, referring to the expression of the agent intention, usually by means of an ACL (Agent Communication Language); conversation level, which concerns to the shared conventions during the interchange of utterances; transportation level that provides mechanisms to help in the transportation of utterances; and the connection level, involving protocols as TCP/IP, IIOP, among others. The conversation protocols can be seen as coordination patterns that restrict the utterance sequences interchanged during a conversation, that is what can be said, by whom and when (Martin et al, 1999). Moreover, an ontology must be defined to establish the possible values for the concepts in the ebXML context and the regulations for the agents. These regulations define the agents' possible behavior and the consequences of their actions (Esteva, 2003).

REFERENCES

Ebxml, 2012. http://www.ebxml.org.

Esteva, M., 2003. Electronic Institutions: from specification to development. *IIIA*, Barcelona, Spain.

Flores, F., 1997. Creando Organizaciones Para El Futuro. *Dolmen Ediciones* S.A. 5ª Edición.

Graham, I., Pollack, N., Smart, A., Williams, R., 2003. Institutionalization of e-Business Standards. *Frontier for Information Systems*.

Hakvoort, R., 2004. ebXML and its impact on conventional Business Information Systems. *1st Twente Student Conference on IT*.

Hofreiter, B., Huemer, C., Klas. W., 2002. ebXML: Status, Research Issues, and Obstacles. In: *Research Issues in Data Engineering: Engineering E-Commerce - E-Business Systems* - RIDE-2EC 2002.

Irani, R., 2002. An Introduction to ebXML. Collaborative Electronic Business is here to stay. In: M. Clark et al, *Web Services Business Strategies and Architectures*. Expert.

Liang, P. et al, 2011. ebXML-Based Electronic Business Interoperability Framework and Test Platform" In: *Electronic Business Interoperability: Concepts, Opportunities and Challenges*, p. 438-456. IGI Global.

Martín, F. J., Plaza, E., Rodriguez-Aguilar, J. A., 1999. An infrastructure for agent-based systems: an interagent approach. *International Journal of Intelligent Systems*.

Paes, R. B., 2005. *Regulando a interação de agentes em sistemas abertos: uma abordagem de leis*. MSc Thesis. Rio de Janeiro: PUC-Rio.

Rawlins, M. C., 2002. *ebXML - A Critical Analysis*, Rawlins EC Consulting.

Smith, D. C., Vinoski, S., 2001. Object Interconnections: CORBA and XML, *C/C++ Users Journal*.

Tidwell, D. Introduction to XML. *IBM developerWorks*.

W3C, 2004a, *Extensible Markup Language (XML)*.

W3C, 2004b, *XML Protocol Working Group*.

Webber, D., Dutton, 2000, A. Understanding ebXML, UDDI, XML/EDI. *XMLGlobal*.

Winograd, T., Flores, F., 1987. *Understanding computers and cognition*. Addison-Wesley.