

# Conducting Business via ebXML

ebXML - The Global Standard for Electronic Business

Stuart **Campbell** <stuart.campbell@TIEglobal.com>

## Abstract

The unique approach of ebXML for application-to-application exchange of business data has been built on a solid experience in business process standardization combined with the opportunities the Internet and XML now offer. This session will provide a technical overview of the ebXML architecture, the approach to messaging and the infrastructure.

## 1. Synopsis

The vision of ebXML is to create a single global electronic marketplace where enterprises of any size and in any geographical location can meet and conduct business with each other through the exchange of XML based messages. In essence ebXML will enable anyone, anywhere, to do business with anyone else.

These high-level, and other identified ebXML requirements, are formalised as a technical architecture which in turn has identified the need for a set of specifications and guidance material that together enable a modular, yet complete, electronic business framework. If the Internet is the information highway for electronic business, then ebXML can be thought of as providing the on ramps, off ramps, and the rules of the road and the technical architecture is thus the highway code.

The objective of this paper is to provide a technical overview of this highway code by discussing:

- Conducting business via ebXML
- ebXML concepts
- Requirements

- Electronic Business
- The Architecture
- ebXML in practice

The current ebXML schedule and intended audience are also discussed finalised by some concluding remarks.

## 2. Conducting Business via ebXML

Before drilling down to the precise ebXML technical detail, it is first beneficial to understand the expectations of implementing a full ebXML business scenario to put everything into context. Enabling this scenario forms the core of the ebXML technical architecture specification.

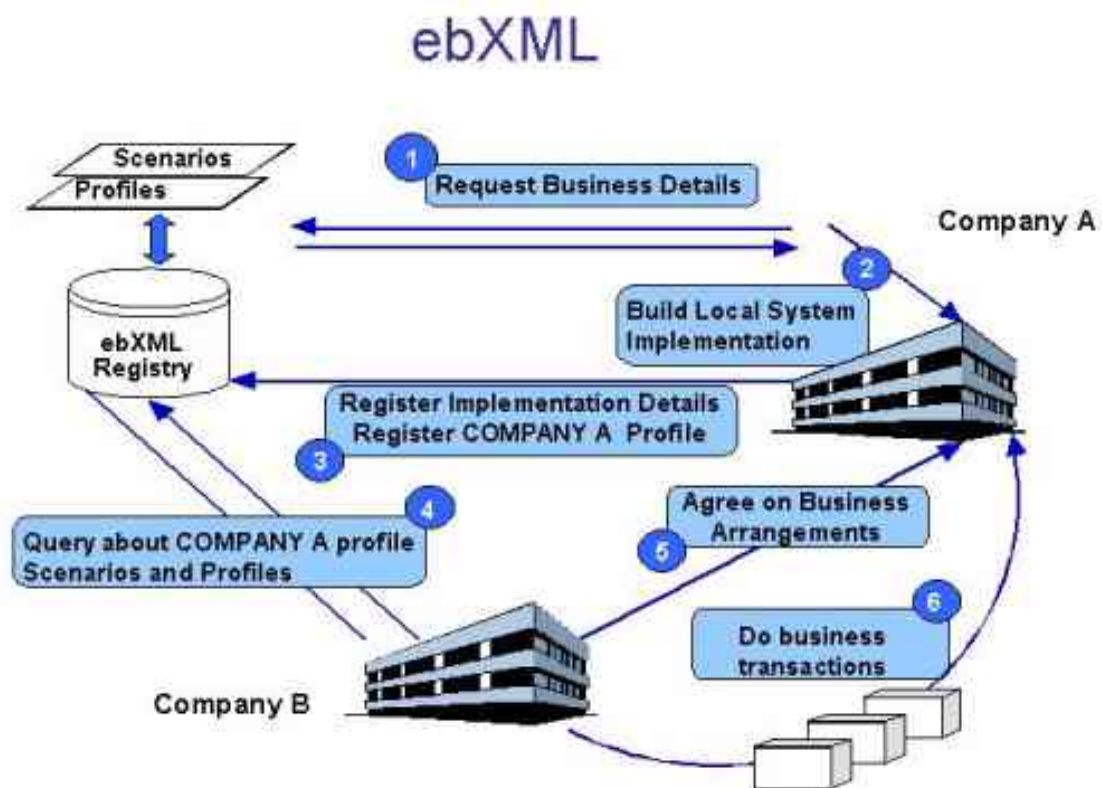


Figure 1. Example Scenario

An example scenario:

1. Company A decides to trade electronically and identifies ebXML as the leading electronic business architecture to enable this. They consult an ebXML registry to examine specifications and example use-case details
2. Company A then implements an ebXML compliant application either by building on top of their current application software, upgrading, using other vendors ebXML compliant modules or by utilising an ebXML compliant shrink-wrapped solution
3. Company A submits, to an ebXML registry, their business profile information which describes their new ebXML related functionality
4. ebXML compliant Company B discovers company A, who is also ebXML compliant, and also identifies the business scenarios supported by company A
5. Company B then sends a request to company A requesting that they engage in a business scenario using ebXML and submits a proposed business arrangement which outlines the proposed business process, business documents, messaging arrangements, security requirements etc. Upon negotiation of this proposal, a collaboration agreement is made between company A and B which forms part of the trading partner agreement
6. Company A and B are now ready to engage in eBusiness using an ebXML conformant messaging

### **3. ebXML Concepts**

As seen above, the main concepts that are established in this implementation scenario are:

- Registry

A "container" for process models, vocabularies, and profiles which may be a networked implementation

- Business Profile

A description of a companies ebXML capabilities and constraints as well as supported business scenarios

- Business Processes

The means and description by which one or more activities are accomplished in performing business - example: Purchasing

- Business Documents

A set of information components which are interchanged as part of a business process - example: Purchase Order

- Trading Partner Agreement

Specifies parameters for businesses to interface with each other

- Messaging

Moves the actual data between trading partners according to an ebXML envelope

## **4. Requirements**

In terms of the requirements for the technical side of ebXML, the following are most pertinent:

### *High-Level Business*

- The need for interoperable global solutions which will accommodate national and international process requirements
- The need to support for both vertical and horizontal solutions regardless of a user's level of sophistication
- The need to support for a range of basic, low cost solutions for Small or Medium Enterprises (SME's) as well as complex solutions for large enterprises

- Relate to the need for a single consistent approach to use XML for B2B (prime focus) and B2C
- Open accessibility
- Leveraging existing technologies and standards
- Integrating with new and legacy systems throughout the enterprise

#### *High-Level Technical*

- Providing a view for integration of business processes among ad-hoc or established independent business partners by electronic means.
- Reducing the need for collaborative business partners to have individual and expensive prior agreement on how to integrate business processes.
- Supporting and representing business processes independent of the technical solution.
- Allowing for both business processes and enabling technologies to evolve independently while retaining long-term investments in both.

#### *Medium-Level Technical*

- The need for an architecture that will ensure and maximize interoperability by supporting common:
  - Business processes
  - Business semantics
- Consistent transport, routing and packaging methodologies to ensure reliable and robust sending and receiving of messages over the Internet
- Enable digital signature and other security related capabilities

#### *eBusiness*

Just as with brick-and-mortar business, in order for enterprises to conduct electronic business with each other they must:

- Discover each other and the products and services they have to offer
- Determine which shared business processes, and associated document exchanges, to use for obtaining products or services from each other
- Determine the contact points and form of communication for the exchange of information
- Agree on the contractual terms of the chosen processes and associated information

And then they can:

- Exchange information and services in an automated fashion in accordance with these agreements

ebXML is designed to meet these needs and is built on three basic concepts:

- Provide an infrastructure that ensures data communication interoperability
- Provide a semantic framework that ensures business interoperability
- Provide a discovery mechanism that allows enterprises to find each other, agree to become trading partners and conduct business with each other

*Infrastructure*, to ensure data communication interoperability, is provided through:

- Standard message transport mechanism with a well defined interface, packaging rules, and a predictable delivery and security model
- A 'business service interface' that handles incoming and outgoing messages at either end of the transport

*Semantic Framework* , to ensure commercial interoperability, is provided through:

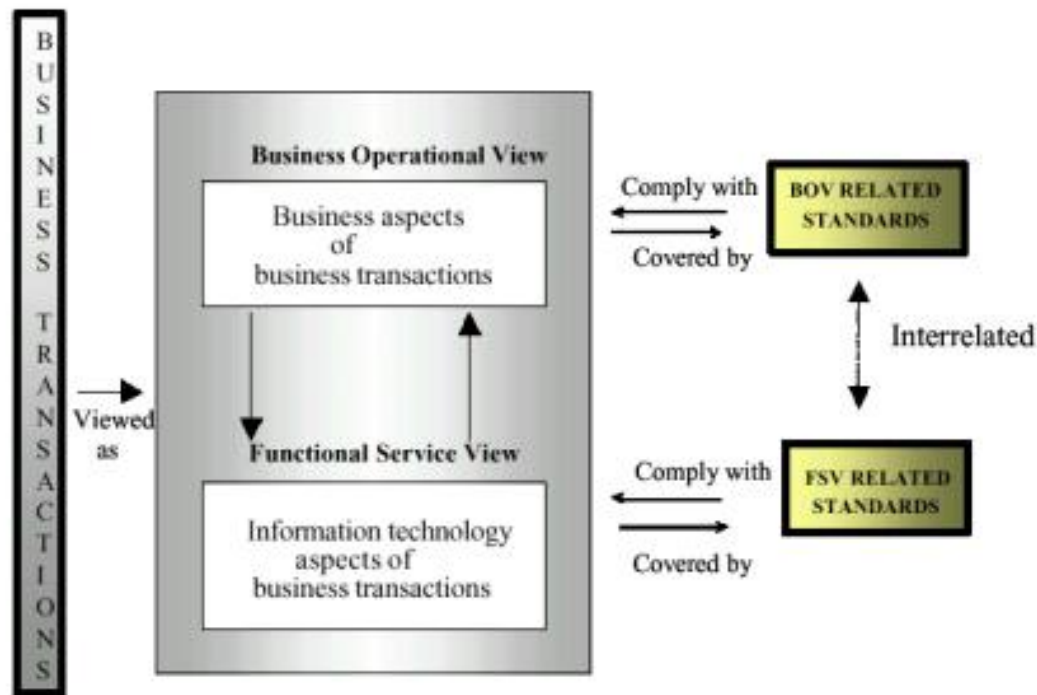
- A metamodel for defining business process and information models
- A set of re-useable core components that reflect common business semantics and XML vocabularies
- Process for defining actual message structures and definitions as they relate to the activities of a Business Process model

*Discovery Mechanism* to allow enterprises to find each other, agree to establish business relationships, and conduct business, is provided through a:

- Shared repository network where enterprises can register and discover each other's business services via partner profile information
- Process for defining and agreeing to a formal Collaboration Protocol Agreement (CPA), if desired, which can be based on the intersection of individual businesses Collaboration Protocol Profiles (CPP)
- Shared repository for company profiles, business process models and related message structures

## **5. The Architecture**

The ebXML Technical Architecture is made up of two basic views based upon the ISO Open-edi reference model.



**Figure 2. Open-edi Model**

- Business Operational View (BOV)

Describes the business processes in a format that is independent from any exchange or programming language. For this purpose, ebXML are using the Unified Modelling Language (UML), and the UN/CEFACT Modelling Methodology (UMM).

- Functional Service View (FSV)

Describes the technical framework used to discover and convey this business information.

## 6. The ebXML Business Operational View (BOV)

Business Knowledge is captured in a registry. The registry contains data and process definitions, including relationships and cross-references, as expressed in business terminology and accessible by classification scheme. The registry is the bridge



between the specific business or industry language and the knowledge expressed by the models in a more generalised industry neutral language.

The first phase defines the requirements artefacts, which describe the problem using Use Case Diagrams and Descriptions. If registry entries are available they will be utilised, otherwise new registry entries will be created.

The second phase (analysis) will create activity and sequence diagrams describing the Business Processes. Class Diagrams will capture the associated information parcels (business messages). The analysis phase reflects the business knowledge contained in the registry. No effort is made to force the Application of object-oriented principles. The class diagram is a free structured data diagram.

The design phase is the last step of standardisation, which may be accomplished by applying object-oriented principles. In addition to generating collaboration diagrams, a state diagram may also be created. The class view diagram from the analysis phase will undergo harmonisation to align it with other models in the same industry and across others.

Therefore in ebXML interoperability is achieved by applying Business Objects across all class models. The content of the Business Object Library is created by analysing existing Business Objects as used by many industries today in conjunction with the registry content and ebXML selected modelling methodology.

## **7. The ebXML Functional Service View (FSV)**

The heart of the FSV is a set of distributed Repositories that are retrieved via ebXML Registries. These registries/repositories hold the Business Process and Information models from the BOV, the ebXML Metamodel, Collaboration Profiles and the ebXML specifications.

The Collaboration Protocol Profiles, CPP, contains the profile a trading partner's IT capabilities, such as transport protocols, security solutions and supported ebXML scenarios.

The Collaboration Protocol Agreements, CPA, are the agreement between two partners on the common profile based on their respective Collaboration Profiles.

Once the CPA is agreed, the business messages, or Payload, can be transmitted between the partners. The physical layers of the communication are defined by the ebXML Messaging Services, which describe the protocol supported, the MIME packaging and the transport header information.

## 8. Outputs

The ebXML architecture defines how ebXML envisions an eBusiness scenario from the original business requirements by identifying:

- Methodologies for defining:
  - Company Profiles
  - Trading Partner Agreements
  - Business processes
  - Business process documents (messages)
  - Common semantics (vocabulary)
- Catalogues of:
  - Common business processes
  - Common (core) semantics
- Specifications for:
  - A uniform message transport layer
  - Repository interfaces for the above catalogues (and other content) which can be used to design, register or discover others business process and trading arrangements
- Accompanying information

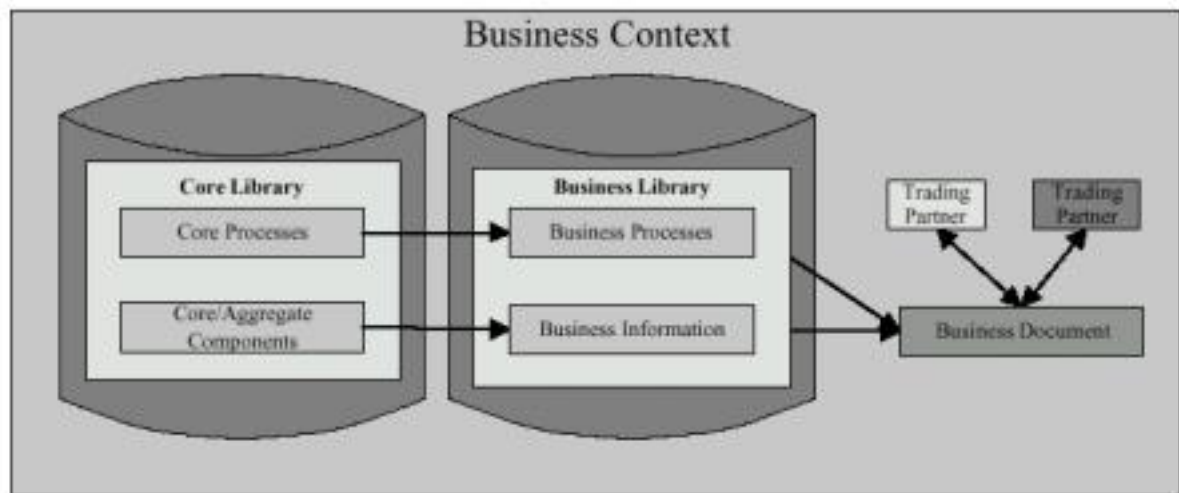
- Overview
- Guidance
- Examples

Broadly speaking, these items then map directly to an integrated suite of ebXML specifications and other material (see annex A) that defines the lower levels of the ebXML architecture. The production of these documents has been assigned to the groups below and most are currently under review either for internal quality control or, as a second stage, for external public review

- Business Process
- Core Components
- Registry and Repository
- Trading Partners
- Transport, Routing and Packaging

These groups are assisted by other groups who are considering requirements, technical architecture, quality and marketing

## **9. Business Process**



**Figure 3. Business Process**

The Business Process models describe how business processes. The specification for business process definition enables an organization to express its business processes in a specific scenario so that they are understandable by other organizations. This enables the integration of business processes within a company, or between companies. The implemented business process may be based upon a context dependant library of business processes, which are in turn based upon a core library of processes

These models are built on the basis of an ebXML metamodel using UML (Unified Modeling Language) and the UMM (UN/CEFACT Modeling Methodology) which is used to define:

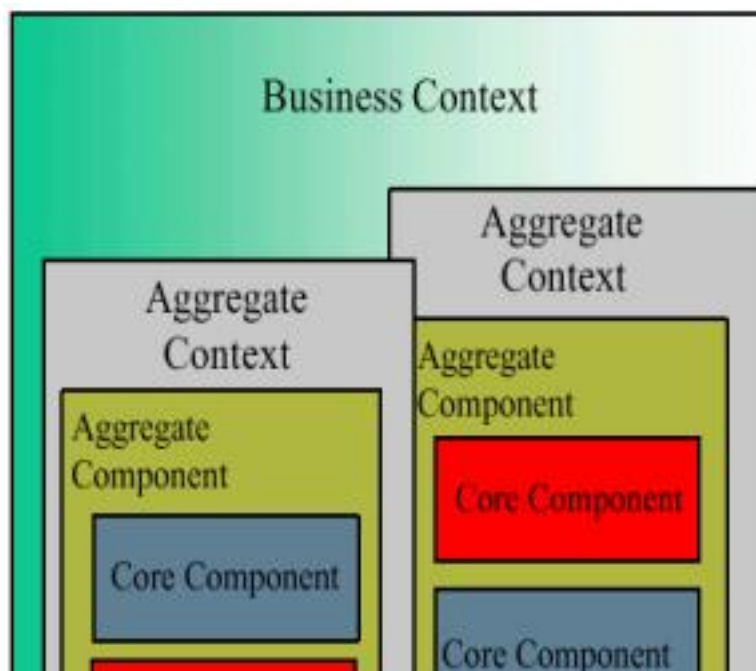
- Semantic subsets
- UML Specification schema
- And an XML representation of such a schema

These allow:

- Define for trading partners

- Role
- Relationship
- Responsibility
- Interaction between
  - Roles
  - Business transactions
  - In relationship with Collaboration Protocol Profiles and agreements (CPP and CPA)

## 10. Core Components



**Figure 4. Core Components**

The core components define reusable components that can be applied in a standard

way within a business context. These Core Components represent the objects of electronic business. They are defined using items that are common across businesses. This enables users to define data that is meaningful to their business while also continuing to maintain interoperability with other business applications. An example of a core component could be a country, address or a party.

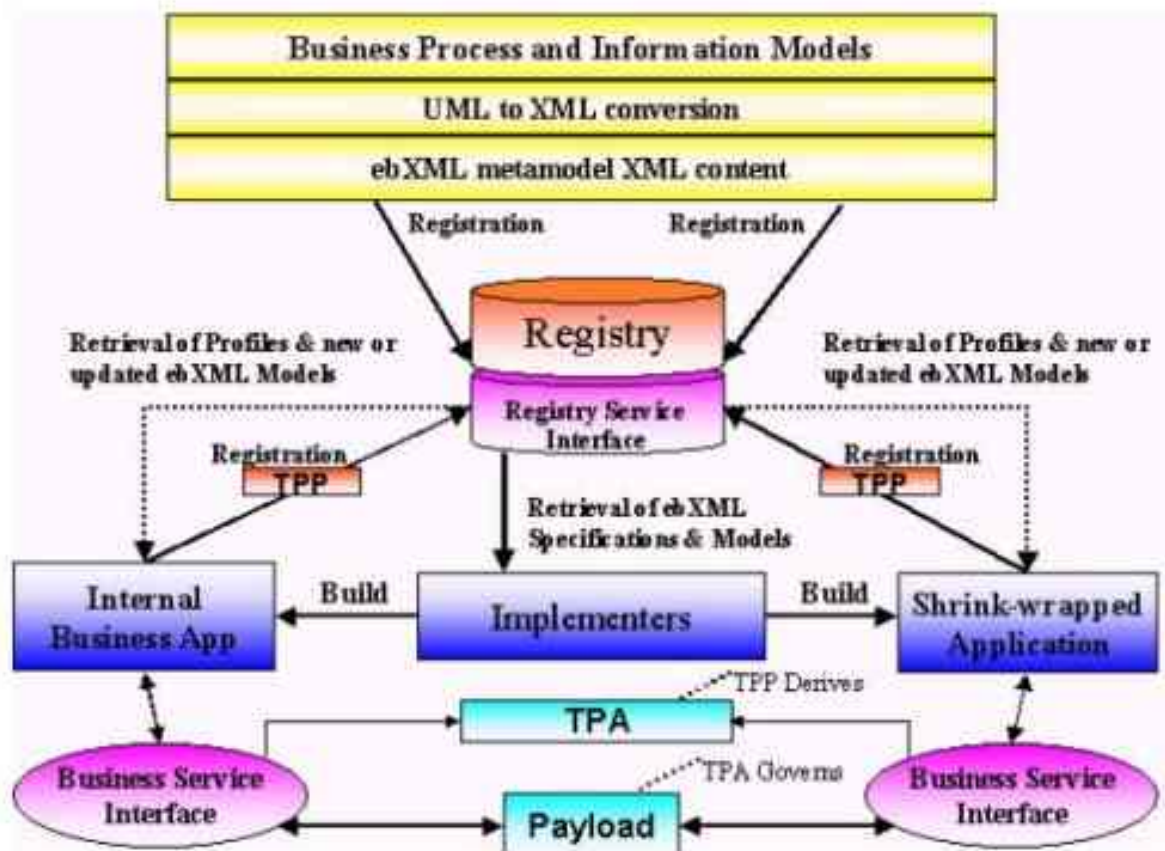
The core components methodology specifications and resultant catalogue are designed to capture:

- Business concepts
- Relationship between business concepts

Core components are:

- Individual pieces (objects) of business information
- Family of business information objects
- Further entries to the ebXML registry
- Capture a minimal set of information to satisfy eBusiness needs
- Capable of containing other core components
- Uniquely identified using Identifiers)
- Facilitate multilingual support

## **11. Registry and Repository**



**Figure 5. Registry and Repository**

The Registry and Repository provides a number of key functions. For the user (application) it stores company profiles and Trading Partner specifications. These give access to specific business processes and information models to allow updates and additions over time. For the application developer it will store not only the final business process definitions, but also a library of core components. When considering the registry it should be clear that this could be a network of ebXML compliant registries and repositories.

In essence the registry will offer:

- Repository of data relationships

- Rules for exchanging information
- Publicly available documents models
- Examples: DTD, schemas...
- Retrieval of profiles
- Secured access - authentication
- Registry of services and clients
- Simple to use globally
- Easily discoverable via internet
- Scalability
- Based on industry standards

## 12. Trading Partner Information



**Figure 6. Trading Partner**

Trading Partner information, through the Collaborative Partner Agreement, defines the technical parameters of the Collaborative Partner Profiles (CPP) and Collaborative Partner Agreements (CPA). These capture critical information for



communications between applications and business processes and also records specific technical parameters for conducting electronic business.

The Collaboration Protocol Profile (CPP) includes:

- Contact information
- Industry classification
- Supported business processes
- Interface requirements
- Messaging services requirements
- Security...

The Collaboration Protocol Agreement (CPA):

- Is the negotiated intersection of two CPPs to be used by all the parties who are the roles in a business process
- Identifies implemented messaging services
- Identifies implemented Business Process Requirements...

## **13. Messaging Services**

The ebXML Messaging Service specification defines the set of services and protocols that enables electronic business applications to exchange data. The specification allows any application-level protocol to be used. These can include common protocols such as SMTP, HTTP, and FTP. Well-established cryptographic techniques can be used to implement strong security. For example, secure protocols such as HTTPS can be used to guarantee confidentiality. In addition, digital signatures can be applied to individual messages or a group of related messages to guarantee authenticity.

## 14. EbXML in practice

Conducting electronic business with ebXML is essentially a step-by-step process, as illustrated in the example below. There are many different scenarios in which these activities may be performed in slightly different sequences and with different scope and focus.

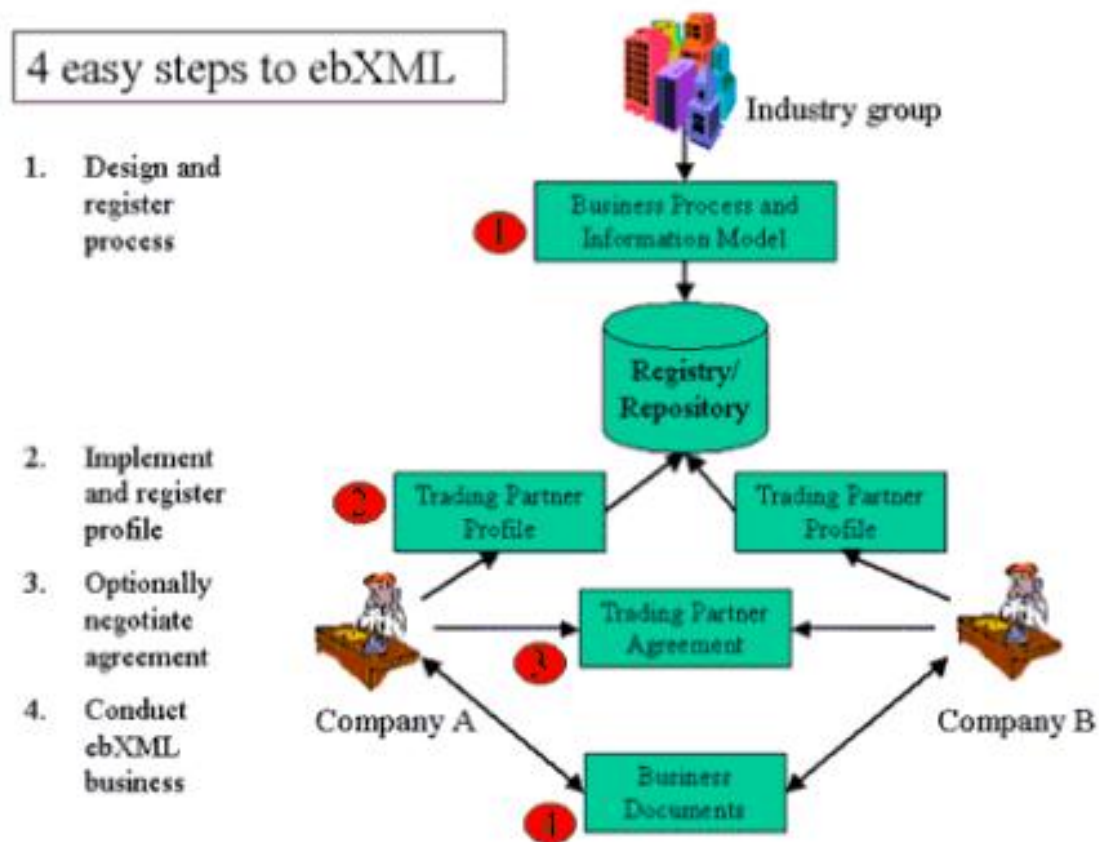


Figure 7. ebXML in practice

1. Design and register business processes and information models

- The implementer browses the repository for appropriate business processes, or for the process the intended partner is registered to support
2. Implement business service interfaces and register Collaborative Partner Profiles
    - The implementer buys, builds, or configures application(s) capable of participating in the selected business process.
    - The implementer registers his (software's) capability to participate, in the form of a Collaborative Partner Profile.
  3. Optionally negotiate and define a Collaborative Partner Agreement (CPA)
    - The two parties negotiate technical details and/or functional overrides, and draw up the result in the form of a CPA
    - Parties optionally register the CPA
  4. Exchanging messages between business partners.
    - The parties (software) send and receive ebXML messages containing ebXML business documents, over the secure and reliable ebXML Messaging Service.

## **15. The schedule**

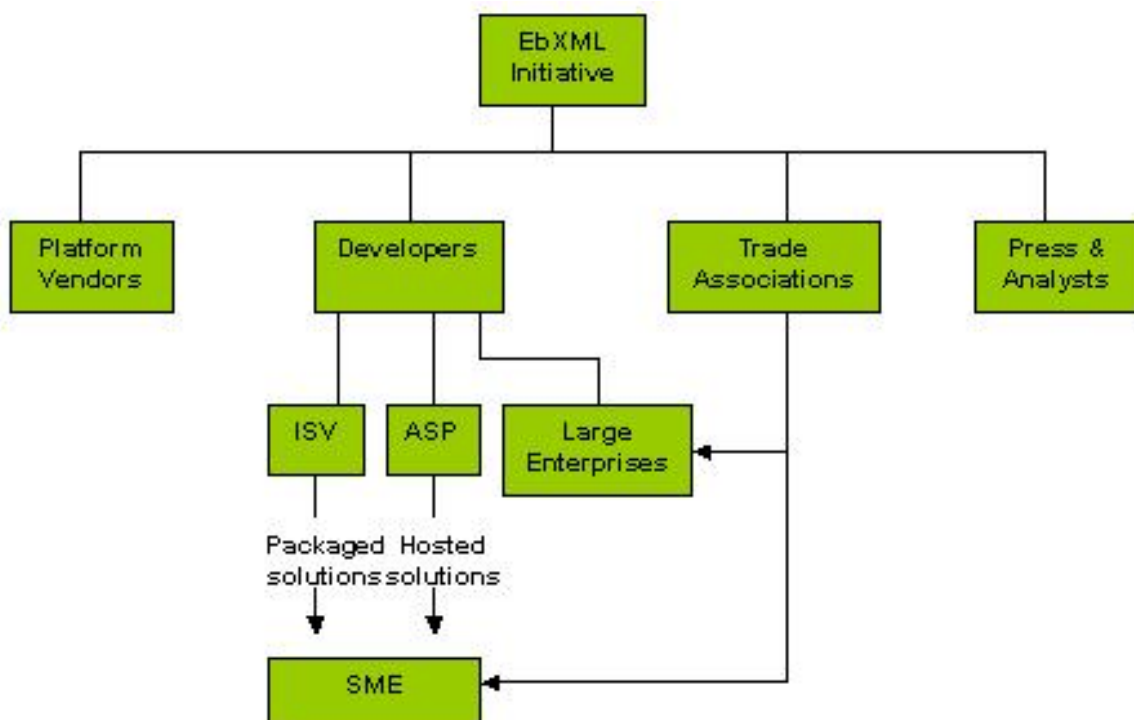
In terms of scheduling of the above activities the following is relevant:

- Identifying user requirements DONE
- Providing an overall architecture DONE
- Identifying / Building architecture components Final delivery May
- Providing guidance material and examples Final delivery May

- Operating proof of concepts Ongoing
- Identifying ebXML conformance Final delivery May

In fact, the initial phase of the ebXML project is completed in May and thus all activity should be complete by the time this paper is issued. We shall see!

## 16. The audience



**Figure 8. ebXML audience**

The targeted audience for the ebXML initiative is not normally the end user, but platform vendors, developers and trade associations. The latter will provide industry specific guidelines of the ebXML specifications. Large enterprises may have the resources to develop their own ebXML solutions, but for small and medium sized enterprises, Independent Software Vendors or Application Software Providers will provide ready to use solutions.

## **17. Conclusion**

The success of ebXML will not be in the proof of abstract theories or the preciseness of microscopic architectural details but, as with all standards, will be determined by its adoption by vendors and thus users. ebXML offers an integrated portfolio of activities and specifications which will undoubtedly enable eBusiness but eBusiness and ebXML does not end here. It is important to get and monitor usage, find the gaps and the errors and fill them in.

It is also important to remember that although XML's key is its extensibility and flexibility even it needs some level of order and as David Orenstein, of Business 2.0., wrote in a recent article:

"Once again standards have prevailed over pettiness on the Web. All along the way, the Web has survived and thrived because standards have always won out over silly vendor infighting. HTML survived, IP survived, XML survived, and perhaps Web services will survive. That means your ability to use the Web to do business can survive too. And you thought the United Nations wasn't able to bring peace."

## **18. Annex A - ebXML Technical Deliverables**

Specifications

<i>Team</i>	<i>Specification Name</i>
Core Components	Rules for application of Context ebXML The role of context in the re-usability of Core Components and Business Processes
	Methodology for developing Core Components ebXML Methodology for the Discovery and Analysis of Core Components
	Naming Conventions - ebXML Naming Conventions for Core Components and Business Processes
	Methodology for Constructing Documents ebXML Specification for the application of XML-based assembly and context rules
Business Process	Specification Schema ebXML Business Process Specification Schema
Trading Partners	Collaboration-Protocol Profile and Agreement Specification
Transport Routing & Packaging	ebXML Message Service Specification
Registry & Repository	Registry Information Model
	Registry Services

**Table 1.**

Reference Documents

<i>Team</i>	<i>Reference Document Name</i>
Technical Architecture	ebXML Glossary
Quality Review	ebXML Roadmap
Core Components	Core Component and Business Process Document Overview
	Core Component Catalog (ongoing)
Business Process	Worksheets for Specification Schema
	Common Business Process Catalog (ongoing)
	E-Commerce Patterns
Security	Security Risk Assessment
Registry & Repository	Registry Information Model

**Table 2.**

## Acknowledgements

Publicly owned company TIE develops and sells advanced, XML/EDI™ -based B2B eCommerce software and services. TIE's software and services solutions enable buyers and product providers to engage in fast and efficient business through "Digital Business Communities" (DBC's) on the Internet. DBC's are Internet portals for companies operating in a vertical market segment within specific regions. TIE sets up the DBC's and ensures that they are well integrated with the internal business systems. The software that establishes the links between the companies or to the DBC is developed in-house at TIE. TIE has a wide clientele in the Benelux and aims to become the market leader in Europe in the immediate future. Its customers come from many market sectors, including retail, healthcare, banking and insurance, transport, production, and food.

## Glossary

CPA

Collaboration Protocol Agreement

CPP	Collaboration Protocol Profiles
UML	Unified Modelling Language
UMM	UN/CEFACT Modelling Methodology

## Biography

### Stuart **Campbell**

Technical Strategy Director  
TIE Holding  
Amsterdam  
The Netherlands  
Email: [stuart.campbell@TIEglobal.com](mailto:stuart.campbell@TIEglobal.com)

*Stuart Campbell* - Stuart Campbell (34). As Technical Strategy Director, Stuart's responsibilities lie in the field of defining, planning, and portraying TIE's future technical direction. This includes working with external partners to build strategic relationships and representation in international, technical, eBusiness forums especially EDI/XML initiatives such as ebXML. Stuart has been involved in the field of eBusiness since 1989 when his career started after graduating from Loughborough University of Technology with a degree in Electronic and Manufacturing Engineering. Initially EDI project manager for ICL, Stuart then moved to manage the technical assessment activities in the Western European EDIFACT Board based within the European Commission. After a period involved with EDI software development he was invited to lead the eCommerce activities in the European Standards Institutes (CEN) European Board for EDI/EC Standardization (EBES). Before joining TIE in January 2001, Stuart worked as Solutions Director for CMASS/Vintura where he managed and was involved in diverse eBusiness projects.