

XML and Electronic Commerce

Web Services Japan 2001
Yokohama, 25 July 2001

Jon Bosak
Sun Microsystems

XML Basics	A-1
Documents and Data.....	B-1
ebXML: Putting Commerce Online.....	C-1
An E-Commerce Roadmap.....	D-1
Getting to Level 2	E-1
Standardizing Business Documents.....	F-1

XML Basics

XML is.....	A-2
XML is "syntax, not semantics"	A-3
XML in one slide.....	A-4
The industrial function of XML.....	A-5
The social function of XML	A-6
What XML accomplishes	A-7

XML is...

- A simplified subset of SGML (ISO 8879:1986) developed by a cross-industry group organized and led by Sun Microsystems
 - Not new: builds on 30 years of research and 14 years of SGML standardization
 - Offers powerful data modeling -- no limits on namespace or structural depth
 - But small enough for Web browsers
- Not a language but a metalanguage
 - Designed to support the definition of an unlimited number of languages for specific industries and applications
 - All XML languages can be processed by a single lightweight parser

XML is "syntax, not semantics"

- Tags have no predefined meaning
- Unlike HTML, XML by itself conveys only content and structure, not presentation, behavior, or meaning
- The meaning of XML languages must be specified outside of XML itself
 - Operational semantics: programs, servlets, applets, scripts, stylesheets...
 - Definitional semantics: prose, namespaces, ontologies, UML diagrams...

XML in one slide

- Legal XML documents are called **well-formed**
- A well-formed document describes a **logical tree**
- If a well-formed document conforms to an optional grammar or schema (e.g., a DTD), it is also **valid**
 - Well-formedness is a property of the **document**
 - Validity is a property of the **relationship** between a document and a grammar or schema

The industrial function of XML

XML provides a standard framework for making agreements about communication.

- Industry DTDs and schemas
- Industry namespaces

But XML does not make those agreements. That would be magic!

Agreements are made by people working together on a shared problem. This is the process of *language standardization*.

The social function of XML

XML has a social agenda: freedom from vendor control.

- (1996) We knew that HTML could not support data exchange in general
- Without XML, HTML would have been replaced with a binary, proprietary format controlled by a single vendor
- The alternative was SGML (international, open, text-based standard, ISO 8879-1986)
- XML put SGML on the Web. Result:
 - Users can define their own data exchange standards
 - There are many inexpensive, robust tools
 - In theory, data belongs to the people who create it rather than to the software vendors
 - In practice, freedom depends on the open standardization of XML tag languages

What XML accomplishes

XML standardizes the **concrete syntax** of data exchange in a **text-based notation** designed to be **obvious** to both people and processes.

Deploying XML creates an **open, standardized information infrastructure**.

1. Standardized parsers for putting data into memory
2. Standardized interfaces (DOM and SAX) for processing the data
3. Standardized ways to display data (CSS, XSL-FO)
4. Standardized ways to query data (XPath, XQuery)
5. Standardized ways to link data (XLink, XPointer)
6. Standardized training of people in both publishing and data processing

Documents and Data

What's a document?	B-2
Separation of data from processing	B-3
The document aspect of XML	B-4
Data into documents	B-5

What's a document?

- A document is data that you can **read**.
- Document requirements are a **superset** of data requirements (e.g., recursion)

The basic problem with documents is that we need to display them in many **different forms**. This is the problem that SGML was originally designed to solve. XML inherits the solution to that problem.

Example: This presentation.

- Written in XML
- HTML generated using a stylesheet for online publishing
- RTF generated using a different stylesheet

Separation of data from processing

- The SGML/XML publishing model decouples data from processing
- This isolates changes in large systems, making them more flexible and reliable
- Basing a system on XML makes it well-suited to transactional processing in a heterogeneous, asynchronous, distributed environment (like the Web)

The document aspect of XML

XML uses documents as the transfer mechanism for data. XML is **text**.

- Creates large, standardized collections of data that can be processed by simple text tools
- Creates an infrastructure in which human beings are still part of the process
 - Troubleshooting
 - Generation of human-readable deliverables (e.g., catalogues)
 - Integration into existing social institutions
- Promotes a message-oriented view of electronic commerce that isolates business transactions from differences in software, hardware, system architectures, and application programming languages

Data into documents

XML was designed to enable many different views to be generated from the same set of data using stylesheets.

So it is perfectly suited to

- Report generation
- Display of business data to consumers
- Display of business data to programmers and business professionals
- Various forms of electronic delivery (e.g., Web sites)

But the markup language must be designed to make this easy.

ebXML: Putting Commerce Online

The ebXML initiative	C-2
The ebXML infrastructure (May 2001)	C-3
E-commerce architecture: ebXML	C-4
Using the infrastructure	C-5
Sun's role in ebXML	C-6

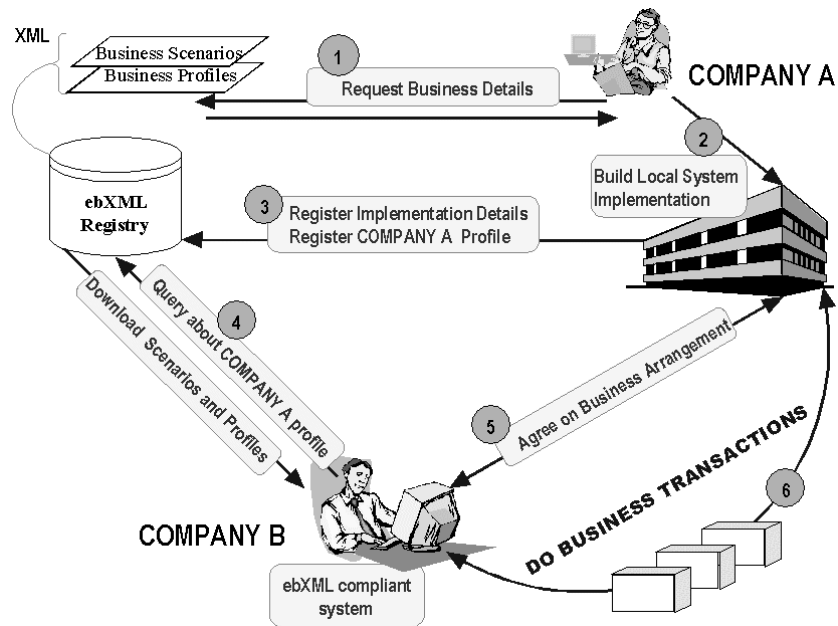
The ebXML initiative

- Joint 18-month effort of OASIS (the Organization for the Advancement of Structured Information Standards) and UN/CEFACT (United Nations Centre for Trade Facilitation and Electronic Business)
- Over 1000 participants from all over the world
- The vision: A global electronic market place where enterprises of any size, anywhere can:
 - Find each other electronically
 - Conduct business through the exchange of XML based messages
- <http://www.ebxml.org/>

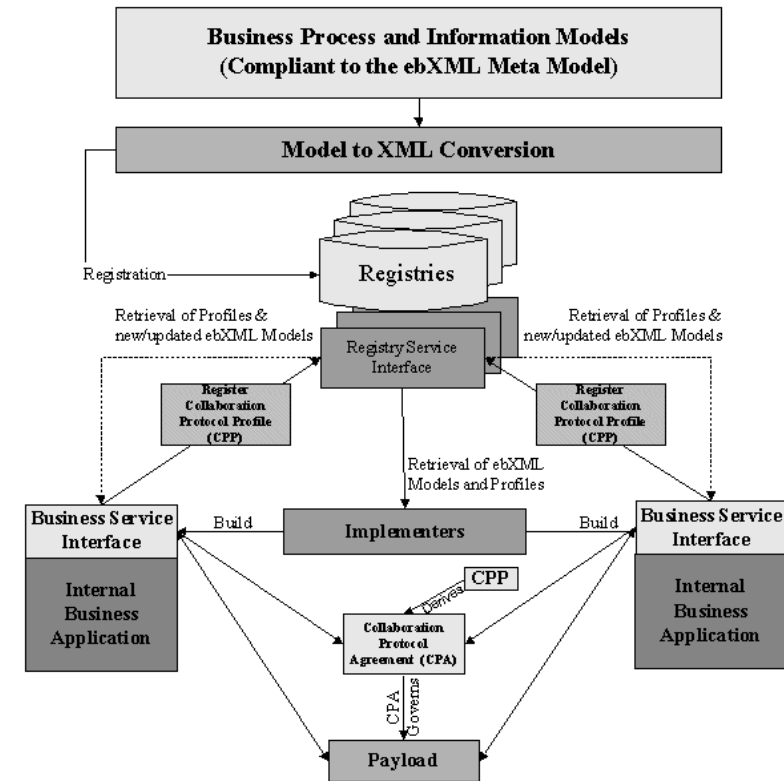
The ebXML infrastructure (May 2001)

- Standard messaging services for XML
- Standard registry/repository for storing business models and profiles
- Standard mechanism for trading partner agreements
- Standard architecture for forming trading partner agreements and exchanging business messages

E-commerce architecture: ebXML



Using the infrastructure



Sun's role in ebXML

- We believe in an open approach to electronic commerce
- We believe that no single company or country should dominate the infrastructure for electronic commerce
- We believe in empowering users
- ebXML is a nonprofit, independent effort sponsored by the United Nations
- ebXML is the only existing non-proprietary framework for electronic commerce
- I am proud to say that Sun has been the single strongest industry supporter of the ebXML effort

Everyone who believes in open standards for electronic commerce should support ebXML.

An E-Commerce Roadmap

Levels of implementation	D-2
Level 1 (you are here).....	D-3
Level 2 (near future)	D-4
Level 3 (distant future)	D-5

Levels of implementation

I believe that the evolution of XML-based electronic commerce will take place in three phases or levels of implementation.

1. Standardize the *infrastructure* (messaging, discovery, trading partner agreements)
2. Standardize the *messages* (business documents: purchase orders, invoices, ...)
3. Standardize *machine-processable semantics* (business process models)

Level 1 (you are here)

With the completion of the ebXML infrastructure specifications in May 2001, messaging, discovery, and trading partner agreements are now standardized and can be automated.

Most primitive implementation: email ad-hoc XML, conducting all other aspects manually using email or phone calls

- This would usually be a one-sided relationship
- But it allows the smallest businesses to participate
- In Level 1, the role of the repository is to supply basic infrastructure specifications and commonly used industry tag sets

Level 2 (near future)

- Level 2 adds standardized XML messages defined by humans within some standardized semantic framework using standardized core components (data elements)
 - Enables off-the-shelf software solutions
 - Enables a transition from existing business practices by replacing traditional business documents with human-readable XML equivalents
- Many options for defining the semantic framework
 - UML
 - Prose
 - Prose + DTD
- In Level 2, the repository will include descriptions of standard business models, standard core components, and standard XML business documents

Level 3 (distant future)

In Level 3, standardized messages are automatically generated from a standardized semantic framework

- Modeling done from the top down
- Schemas generated automatically
- The whole system operates without human intervention
- The repository now includes business models designed for automation
- Advantages:
 - Completely automated
 - Allows dynamic modification of business models
- Problems to be solved:
 - Development of the technology itself
 - Entry point for SMEs
 - Transition from existing business and legal systems

Getting to Level 2

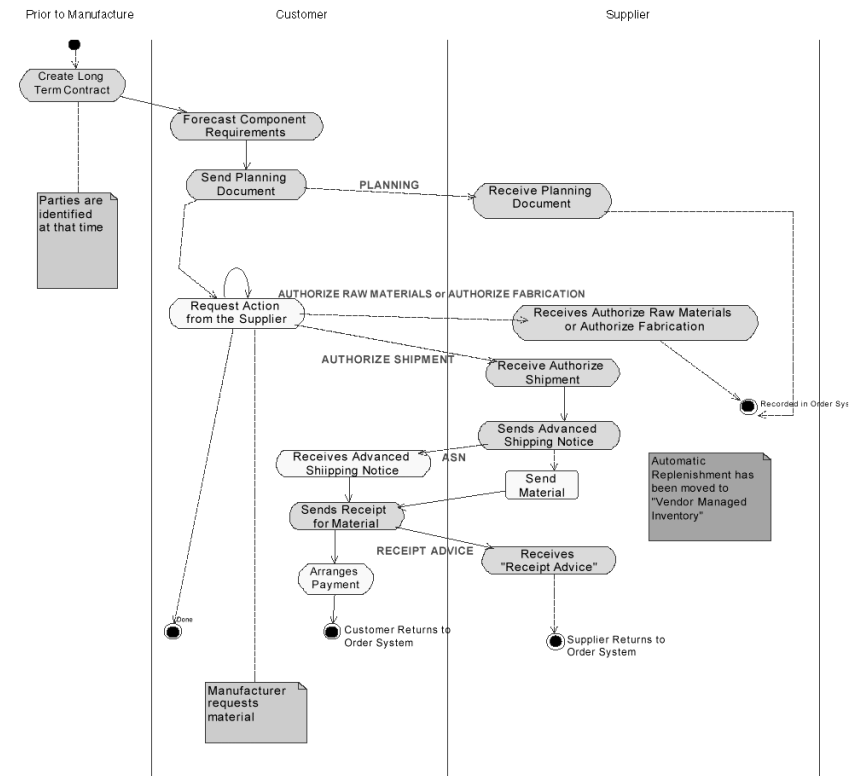
Review: Levels of implementation	E-2
Level 2 example: materials management.....	E-3
Constructing a business language	E-5
Document components	E-6
Context drivers.....	E-7
Context example: address	E-8

Review: Levels of implementation

1. Standardize the *infrastructure* (messaging, discovery, trading partner agreements)
 - The ebXML infrastructure provides this level of functionality
2. Standardize the *messages* (business documents: purchase orders, invoices, ...)
 - This is the next stage of development
3. Standardize *machine-processable semantics* (business process models)
 - This is a long-term project for the future

Level 2 example: materials management

In level 2, standardized business documents take their meaning from standardized business models. This diagram shows the role of planning and shipping documents in a business model from the automotive industry.



Constructing a business language

To achieve the next level of e-commerce implementation, we need to standardize a common business language. Examples of what needs to be standardized:

- Date
- Weight
- Price
- Address
- Telephone number
- Party
- Purchase order
- Shipping notice
- Invoice

Document components

- Atomic components
 - Date
 - Weight
 - Price
- Aggregate components
 - Address
 - Telephone number
- Core blocks
 - Party
- Documents
 - Purchase order
 - Shipping notice
 - Invoice

Context drivers

"Standard" data components change structure when put into different contexts.

- Business process
- Industry
- Product type
- Geographical region
- Regulatory environment
- Role (vendor, customer, etc.)

Context example: address

Even such a simple data item as an address will change depending on the context.

- Addresses in Japan are different from addresses in the United States
- Addresses in the auto industry are different from addresses in other industries

Standardizing Business Documents

The immediate problem	F-2
A plan for averting chaos	F-3
Requirements for a Universal Business Language (UBL)	F-4
My proposal: start with xCBL 3.0	F-5
Basic concept	F-6
Possible venues for standardization	F-7
Status	F-8

The immediate problem

Users are rushing to adopt various existing XML business syntaxes.

- cXML
- xCBL
- RosettaNet
- OAG
- BizTalk
- various other dialects

The existence of multiple business languages is a serious interoperability problem. But once adopted, these existing vocabularies will be very hard to dislodge.

A plan for averting chaos

1. Put one of the *existing* XML b2b libraries into a genuine standards process.
2. Populate the process with XML and business domain experts.
3. Persuade other groups working to standardize XML business grammars to join the common effort.
4. Modify the default library as necessary to achieve consensus.
5. Lead users into adopting the standard "Universal Business Language" (UBL)
6. Evolve the standard UBL as we gain experience.

Requirements for a Universal Business Language (UBL)

- Optimized for Internet b2b
- International in scope
- Applicable across any sector or domain of electronic trade, transport, and administration (purchasing, payments, logistics, transportation, statistical reporting, social administration, healthcare, etc.)
- Interoperable with existing EDI systems
- Based on a core library
- Unencumbered by intellectual property claims
- Already proven to work in actual electronic marketplaces
 - Proven implementability
 - Easy testability

My proposal: start with xCBL 3.0

1. xCBL 3.0 represents four years of work that we don't want to start over from scratch.
2. xCBL's IP is unencumbered. *This is really important.*
3. xCBL 3.0 reflects experience gained in EDI, RosettaNet, OBI, and ebXML.
4. Microsoft and SAP are basing systems on xCBL.
5. Many vendors already have xCBL code for use in adapters.
6. The owners of xCBL are willing to put it into a real standards process for the benefit of the industry and to work within that process to advance the industry standard.

In the software business, this is as good as it gets.

Basic concept

The basic idea is to review the baseline syntax line by line in harmony with semantic definition work taking place in the United Nations and to make whatever modifications are necessary in light of experience with other syntaxes:

- X12 and EDIFACT
- RosettaNet
- OAG
- cXML
- Bolero
- et cetera

The details of this process need to be defined in committee.

Possible venues for standardization

- UN/CEFACT (UN/EDIFACT)
 - International participation and credibility
 - Links to major corporate users
 - Links to international organizations (WTO, WCO, IMF ...)
 - But subject to delays and political problems
- OASIS
 - Nonprofit corporation founded in 1993
 - Represents XML vendors and users
 - Excellent, fast, fair committee process
 - Any interested person can join

Status

[Status as of 25 July 2001]

- Sun, Commerce One, SAP, and other software vendors have formed a UBL Organizing Committee
- We are attempting to form a working group in UN/CEFACT to carry out the work of standardization
- We are hoping to begin work after the UN/EDIFACT working group meeting in September

[Status as of 3 October 2001]

- Pending a reorganization in progress in UN/CEFACT, the UBL work will take place in an OASIS technical committee
- The first UBL TC meeting is scheduled for the week of 29 October through 1 November 2001
- For more details: <http://oasis-open.org/committees/ubl/>