Introduction

"There's wind, and then there's a typhoon. In this business, you always have winds. But a 10X force is a change in an element of one's business—a typhoon. Is the Internet a typhoon, or a bit of wind? Is it a force that fundamentally alters our business?"
—Andrew Grove, Chairman and founder, Intel

ebXML is an emerging e-commerce standard that leverages the flexibility of the Extensible Markup Language (XML) to build e-commerce infrastructure. XML is a markup language used to create data and documents for application communications and storage. The "eb" in ebXML stands for "electronic business," and the phrase is pronounced as simply "ee-bee-ex-em-el." In this book we will discuss how ebXML fundamentally changes the way information technology (IT) handles online business transactions.

The ebXML standard is the result of a joint international initiative of the United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT) and the Organization for the Advancement of Structured Information Standards (OASIS). ebXML is a concerted effort to combine the best of two existing standards: electronic data interchange (EDI) and extensible markup language (XML). Prior to ebXML, members of both EDI and XML groups disagreed over details but wanted something from the other side. XML users wanted access to the EDI community's vast inventory of business semantics and standards development, and the EDI community wanted to use XML technology. Joining forces in the ebXML initiative was a win-win situation. Together they set the following goal, according to the published ebXML vision on the organization's Web site:

"The vision of ebXML is to enable a 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."

Why Is ebXML Important?

Why are standards important in e-commerce? The following analogy should help demonstrate. When the home video industry started, movies were available in two popular formats, VHS and Beta. The VHS video format was less expensive, but the Beta offered a higher-quality format. Eventually, however, VHS won out over Beta. Why? Because VCR manufacturers adopted VHS, which—unlike Beta—was not controlled by Sony, one of their competitors. This led to a lower cost for VHS players and cassettes. People that bet on Beta lost out.

Industry standards like VHS and Beta help coordinate the behavior of market players to make wise investments, whether it is a 16-year-old buying the latest Hollywood release or a videocassette player manufacturer building a multimillion-dollar production factory.

The story did not end there, because today there are still more formats for movies. Today we can also purchase movies on DVD or download them as an MPEG on the Internet. New standards come into play in the marketplace, and the market players such as consumers, manufacturers, distributors, and retailers decide whether the standards thrive or die.
How does ebXML fit into this? Technically, ebXML is not a standard yet, but it stands a good chance of becoming an important electronic commerce standard. (See the detailed discussion on a standard versus specifications in Chapter 1). Each year billions of dollars are at stake in business-to-business (B2B) transactions. Companies large

and small have to invest in infrastructure to enable trading of goods and services between them. As a whole, industry players decide on the common architecture and applications for exchanging trading information.

ebXML is a uniform solution for building the e-commerce infrastructure. Instead of diverging proprietary standards, ebXML unites the competing factions under a common banner of international trade. The battleground for ebXML is the backdrop of existing standards such as electronic data interchange (EDI), as well as newer trends, such as the Internet and B2B.

Who Should Read This Book

This book is intended to help technology, business, and sales professionals understand ebXML and how it can be applied to meet business requirements. The goal is to help readers quickly get a grasp on what is involved, whether your role is selling, buying, planning, or implementing a project involving ebXML. The book assumes no previous knowledge of ebXML, XML, or other programming languages, but it is helpful if you have a technical background in IT.

You may or may not already be familiar with ebXML. For those just starting out, this book will give you a sound understanding of the basic concepts and principles, a look at what others are doing, and an understanding of the business and technical solutions that ebXML can address. In short, you will have the background information you need to ask serious questions about ebXML and to plan a project involving ebXML.

For those who are more advanced in their knowledge of ebXML, the companion Web site for the book will help with mastering advanced technical topics and provide additional programming code and reference information for your project.

How the Book Is Organized

There is no way one book (even one filling several volumes) could poke into every nook and cranny of an evolving area as complicated as ebXML. Even if we could lay out all the little pieces in one place, it would be obsolete within a few months. Thus, this book is designed to provide a broad overview, which while fluid, does not change nearly as quickly, so we will have a bare tree on which to hang the leaves of our own experience. It is easier to pick up the details from Web sites, trade publications, and industry conferences than it is to learn the structural overview, because few people have the time and patience to sit down and discuss it. For some of the ebXML resources, you can check out the Resources appendix.
This book is divided into two major sections: Part 1 provides an overview of ebXML, including the basic foundations built on EDI and XML; Part 2 provides a detailed discussion of components of the ebXML architecture, such as business process, core components, messaging, registry, and collaboration protocol agreement and profile.

Part 1: E-Commerce Basics
Chapter 1 provides the business context for ebXML within e-commerce and B2B. We look at how companies can benefit from B2B. We also look at standards, why we need them, and ebXML as an emerging standard versus a set of specifications.
Chapter 2 provides the technical context for ebXML within EDI and XML. What is EDI and XML? We explore basic concepts of XML such as format, namespaces, document

type definitions (DTDs), as well as work through some XML examples to learn how to use XML.
Chapter 3 provides a technical context for ebXML within Web services, which are interactive services built using XML- and HTTP-based technologies. We define Web services and explore how to use it. Web services technology includes languages such as XML and Web services Definition Language (WSDL); protocols such as Simple Object Access Protocol (SOAP) and Universal Description, Discovery, and Integration (UDDI); and enterprise XML frameworks such as the Microsoft-driven BizTalk initiative and RossettaNet.

Part 2: ebXML Technologies
Chapter 4 provides an overview of the ebXML architecture. First, we discuss the business and technical rationale for the overall architecture, framework, and applications. We look at the specific technical pieces such as the business process model, core components, messaging, registry, and collaboration protocol.
Chapter 5 discusses the ebXML business process model. We discuss the business process model in ebXML and how to use it with some examples of documents, transactions, and collaborations.
Chapter 6 looks at the core components of ebXML. We discuss the component architecture and how to use it with some examples of context and assembly rules to put together core components.
Chapter 7 covers ebXML messaging. We discuss what messaging is in an ebXML context and how to use it to send messages between trading partners.
Chapter 8 reviews the ebXML registry. We discuss the registry architecture and its usage with some examples of browsing and querying a registry.
Chapter 9 provides an overview of the ebXML collaboration protocol. We discuss using the protocol as a handshake between partners and show this process using examples of the collaboration protocol agreement and profile.
Chapter 10 shows you how to apply ebXML in your projects. We discuss how to apply the concepts in real-world situations to gather user requirements and manage stakeholder expectations.

Appendixes

The appendixes include listings of ebXML-related resources, references, and a glossary of ebXML terms. The glossary includes a brief explanation of ebXML technical jargon, and it could be useful for navigating through technical discussions. The lists of resources and references serve as a starting point for further research on your own.

About the Web Site

The companion Web site for this book (www.wiley.com/compbooks/chiu) serves as an information resource for further investigation into ebXML. The companion Web site includes references to the ebXML specifications, sample code, demos, and discussion groups.

Detailed technical topics and code examples are posted on the Web site. What language and middleware will we use? ebXML is the business rules for how two different systems talk to each other. Those systems need to be written using a specific application programming language (XML, Java, C, C++, Visual Basic, and so on), executed in a specific middleware (J2EE and COM+), and designed using a specific modeling language (UML, UMM, and UBL).

For a focused presentation without technical clutter, we will use XML as the sole technical language of choice. This philosophy of XML minimalism means you do not