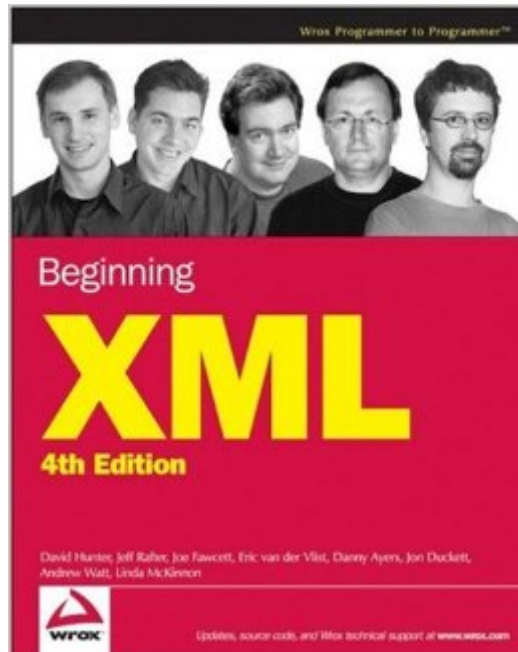


The book was found

# Beginning XML, 4th Edition



## Synopsis

When the first edition of this book was written, XML was a relatively new language but already gaining ground fast and becoming more and more widely used in a vast range of applications. By the time of the second edition, XML had already proven itself to be more than a passing fad, and was in fact being used throughout the industry for an incredibly wide range of uses. With the third edition, it was clear that XML was a mature technology, but more important, it became evident that the XML landscape was dividing into several areas of expertise. Now in this edition, we needed to categorize the increasing number of specifications surrounding XML, which either use XML or provide functionality in addition to the XML core specification. So what is XML? It's a markup language, used to describe the structure of data in meaningful ways. Anywhere that data is input/output, stored, or transmitted from one place to another, is a potential fit for XML's capabilities. Perhaps the most well-known applications are web-related (especially with the latest developments in handheld web accessâ for which some of the technology is XML-based). However, there are many other non-web-based applications for which XML is usefulâ for example, as a replacement for (or to complement) traditional databases, or for the transfer of financial information between businesses. News organizations, along with individuals, have also been using XML to distribute syndicated news stories and blog entries. This book aims to teach you all you need to know about XMLâ what it is, how it works, what technologies surround it, and how it can best be used in a variety of situations, from simple data transfer to using XML in your web pages. It answers the fundamental questions: \* What is XML? \* How do you use XML? \* How does it work? \* What can you use it for, anyway? This book is for people who know that it would be a pretty good idea to learn XML but aren't 100 percent sure why. You've heard the hype but haven't seen enough substance to figure out what XML is and what it can do. You may be using development tools that try to hide the XML behind user interfaces and scripts, but you want to know what is really happening behind the scenes. You may already be somehow involved in web development and probably even know the basics of HTML, although neither of these qualifications is absolutely necessary for this book. What you don't need is knowledge of markup languages in general. This book assumes that you're new to the concept of markup languages, and we have structured it in a way that should make sense to the beginner and yet quickly bring you to XML expert status. The word "Beginning" in the title refers to the style of the book, rather than the reader's experience level. There are two types of beginner for whom this book is ideal: \* Programmers who are already familiar with some web programming or data exchange techniques. Programmers in this category will already understand some of the concepts discussed here, but you will learn how you can incorporate XML technologies to enhance

those solutions you currently develop. \* Those working in a programming environment but with no substantial knowledge or experience of web development or data exchange applications. In addition to learning how XML technologies can be applied to such applications, you will be introduced to some new concepts to help you understand how such systems work. The subjects covered in this book are arranged to take you from novice to expert in as logical a manner as we could. This Fourth Edition is structured in sections based on various areas of XML expertise. Unless you are already using XML, you should start by reading the introduction to XML in Part I. From there, you can quickly jump into specific areas of expertise, or, if you prefer, you can read through the book in order. Keep in mind that there is quite a lot of overlap in XML, and that some of the sections make use of techniques described elsewhere in the book. \* The book begins by explaining what exactly XML is and why the industry felt that a language like this was needed. \* After covering the why, the next logical step is the how, so it shows you how to create well-formed XML. \* Once you understand the whys and hows of XML, you'll go on to some more advanced things you can do when creating your XML documents, to make them not only well formed, but valid. (And you'll learn what "valid" really means.) \* After you're comfortable with XML and have seen it in action, the book unleashes the programmer within and looks at an XML-based programming language that you can use to transform XML documents from one format to another. \* Eventually, you will need to store and retrieve XML information from databases. At this point, you will learn not only the state of the art for XML and databases, but also how to query XML information using an SQL-like syntax called XQuery. \* XML wouldn't really be useful unless you could write programs to read the data in XML documents and create new XML documents, so we'll get back to programming and look at a couple of ways that you can do that. \* Understanding how to program and use XML within your own business is one thing, but sending that information to a business partner or publishing it to the Internet is another. You'll learn about technologies that use XML that enable you to send messages across the Internet, publish information, and discover services that provide information. \* Since you have all of this data in XML format, it would be great if you could easily display it to people, and it turns out you can. You'll see an XML version of HTML called XHTML. You'll also look at a technology you may already be using in conjunction with HTML documents called CSS. CSS enables you to add visual styles to your XML documents. In addition, you'll learn how to design stunning graphics and make interactive forms using XML. \* Finally, the book ends with a case study, which should help to give you ideas about how XML can be used in real-life situations, and which could be used in your own applications. This book builds on the strengths of the earlier editions, and provides new material to reflect the changes in the XML landscape— notably XQuery, RSS and

Atom, and AJAX. Updates have been made to reflect the most recent versions of specifications and best practices throughout the book. In addition to the many changes, each chapter has a set of exercise questions to test your understanding of the material. Possible solutions to these questions appear in Appendix A.

**Part I: Introduction:** The introduction is where most readers should begin. The first three chapters introduce some of the goals of XML as well as the specific rules for constructing XML. Once you have read this part you should be able to read and create your own XML documents.

**Chapter 1: What Is XML?:** This chapter covers some basic concepts, introducing the fact that XML is a markup language (a bit like HTML) whereby you can define your own elements, tags, and attributes (known as a vocabulary). You'll see that tags have no presentation meaning—they're just a way to describe the structure of the data.

**Chapter 2: Well-Formed XML:** In addition to explaining what well-formed XML is, we offer a look at the rules that exist (the XML 1.0 and 1.1 Recommendations) for naming and structuring elements—you need to comply with these rules in order to produce well-formed XML.

**Chapter 3: XML Namespaces:** Because tags can be made up, you need to avoid name conflicts when sharing documents. Namespaces provide a way to uniquely identify a group of tags, using a URI. This chapter explains how to use namespaces.

**Part II: Validation:** In addition to the well-formedness rules you learn in Part I, you will most likely want to learn how to create and use different XML vocabularies. This Part introduces you to DTDs, XML Schemas, and RELAX NG: three languages that define custom XML vocabularies. It also shows you how to utilize these definitions to validate your XML documents.

**Chapter 4: Document Type Definitions:** You can specify how an XML document should be structured, and even provide default values, using Document Type Definitions (DTDs). If XML conforms to the associated DTD, it is known as valid XML. This chapter covers the basics of using DTDs.

**Chapter 5: XML Schemas:** XML Schemas, like DTDs, enable you to define how a document should be structured. In addition to defining document structure, they enable you to specify the individual datatypes of attribute values and element content. They are a more powerful alternative to DTDs.

**Chapter 6: RELAX NG:** RELAX NG is a third technology used to define the structure of documents. In addition to a new syntax and new features, it takes the best from XML Schemas and DTDs, and is therefore very simple and very powerful. RELAX NG has two syntaxes; both the full syntax and compact syntax are discussed.

**Part III: Processing:** In addition to defining and creating XML documents, you need to know how to work with documents to extract information and convert it to other formats. In fact, easily extracting information and converting it to other formats is what makes XML so powerful.

**Chapter 7: XPath:** The XPath language is used to locate sections and data in the XML document, and it's important in many other XML technologies.

**Chapter 8: XSLT:** XML can be transformed into other XML

documents, HTML, and other formats using XSLT stylesheets, which are introduced in this chapter.

**Part IV: Databases:** Creating and processing XML documents is good, but eventually you will want to store those documents. This section describes strategies for storing and retrieving XML documents and document fragments from different databases. **Chapter 9: XQuery, the XML Query Language:** Very often, you will need to retrieve information from within a database. XQuery, which is built on XPath and XPath2, enables you to do this in an elegant way. **Chapter 10: XML and Databases:** XML is perfect for structuring data, and some traditional databases are beginning to offer support for XML. This chapter discusses these, and provides a general overview of how XML can be used in an n-tier architecture. In addition, new databases based on XML are introduced.

**Part V: Programming:** At some point in your XML career, you will need to work with an XML document from within a custom application. The two most popular methodologies, the Document Object Model (DOM) and the Simple API for XML (SAX), are explained in this part. **Chapter 11: The Document Object Model (DOM):** Programmers can use a variety of programming languages to manipulate XML using the Document Object Model's objects, interfaces, methods, and properties, which are described in this chapter. **Chapter 12: Simple API for XML (SAX):** An alternative to the DOM for programmatically manipulating XML data is to use the Simple API for XML (SAX) as an interface. This chapter shows how to use SAX and utilizes examples from the Java SAX API.

**Part VI: Communication:** Sending and receiving data from one computer to another is often difficult, but several technologies have been created to make communication with XML much easier. This part discusses RSS and content syndication, as well as web services and SOAP. This edition includes a new chapter on Ajax techniques. **Chapter 13: RSS, Atom, and Content Syndication:** RSS is an actively evolving technology that is used to publish syndicated news stories and website summaries on the Internet. This chapter not only discusses how to use the different versions of RSS and Atom, it also covers the future direction of the technology. In addition, it demonstrates how to create a simple newsreader application that works with any of the currently published versions. **Chapter 14: Web Services:** Web services enable you to perform cross-computer communications. This chapter describes web services and introduces you to using remote procedure calls in XML (using XML-RPC and REST), as well as giving you a brief look at major topics such as SOAP. Finally, it breaks down the assortment of specifications designed to work in conjunction with web services. **Chapter 15: SOAP and WSDL:** Fundamental to XML web services, the Simple Object Access Protocol (SOAP) is one of the most popular specifications for allowing cross-computer communications. Using SOAP, you can package up XML documents and send them across the Internet to be processed. This chapter explains SOAP and the Web Services Description Language

(WSDL) that is used to publish your service. Chapter 16: Ajax: Ajax enables you to utilize JavaScript with web services and SOAP, or REST communications. Additionally, Ajax patterns can be used within web pages to communicate with the web server without refreshing. This chapter is new to the Fourth Edition. Part VII: Display: Several XML technologies are devoted to displaying the data stored inside of an XML document. Some of these technologies are web-based, and some are designed for applications and mobile devices. This part discusses the primary display strategies and formats used today. Chapter 17: Cascading Style Sheets (CSS): Website designers have long been using Cascading Style Sheets (CSS) with their HTML to easily make changes to a website's presentation without having to touch the underlying HTML documents. This power is also available for XML, enabling you to display XML documents right in the browser. Or, if you need a bit more flexibility with your presentation, you can use XSLT to transform your XML to HTML or XHTML and then use CSS to style these documents. Chapter 18: XHTML: XHTML is a new version of HTML that follows the rules of XML. This chapter discusses the differences between HTML and XHTML, and shows you how XHTML can help make your sites available to a wider variety of browsers, from legacy browsers to the latest browsers on mobile phones. Chapter 19: Scalable Vector Graphics (SVG): Do you want to produce a custom graphic using XML? SVG enables you to describe a graphic using XML-based vector commands. This chapter teaches you the basics of SVG and then dives into a more complex SVG-based application that can be published to the Internet. Chapter 20: XForms: XForms are XML-based forms that can be used to design desktop applications, paper-based forms, and of course XHTML-based forms. This chapter demonstrates both the basics and some of the more interesting uses of XForms. Part VIII: Case Study: Throughout the book you'll gain an understanding of how XML is used in web, business-to-business (B2B), data storage, and many other applications. The case study covers an example application and shows how the theory can be put into practice in real-life situations. The case study is new to this edition. Chapter 21: Case Study: Payment Calculator: This case study explores some of the possibilities and strategies for using XML in your website. It includes an example that demonstrates a loan payment calculator by creating a web page using XHTML and CSS, communicating with a local web service using AJAX, utilizing an XML Schema to build data structures in .NET, and ultimately using the Document Object Model to display the results in SVG. An online version of this case study on the book's website covers the same material using Ruby on Rails instead of .NET. Appendixes: Appendix A provides answers to the exercise questions that appear throughout the book. The remaining appendixes provide reference material that you may find useful as you begin to apply the knowledge gained throughout the book in your own applications. These are: Appendix B: XPath

Reference; Appendix C: XSLT Reference; Appendix D: The XML Document Object Model; Appendix E: XML Schema Element and Attribute Reference; Appendix F: XML Schema Datatypes Reference; Appendix G: SAX 2.0.2 Reference. Appendixes A, B, and C are included within the book; Appendixes Dâ€”G are available on the book's website.

## Book Information

Paperback: 1080 pages

Publisher: Wrox; 4 edition (May 21, 2007)

Language: English

ISBN-10: 0470114878

ISBN-13: 978-0470114872

Product Dimensions: 7.6 x 2.1 x 9.5 inches

Shipping Weight: 2.5 pounds

Average Customer Review: 3.6 out of 5 starsÂ Â  See all reviewsÂ  (19 customer reviews)

Best Sellers Rank: #588,914 in Books (See Top 100 in Books) #91 inÂ Books > Computers & Technology > Programming > Languages & Tools > XML #788 inÂ Books > Textbooks > Computer Science > Software Design & Engineering #1667 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Development

## Customer Reviews

Hello everybody. Having more than 14 yrs experience in all kind of IT techies, I always had a problem when it came to XML: too big of a confusing terminology and concepts that were difficult to understand. Aside from being a markup language used for structuring data I did not know too much. I plunged into all sort of documentation available on the web, browsed a book or two either from my friends' bookshelves or in a bookshop, but stil questions like "what is the difference between a DTD and a schema" or "should I use URL or URN for a namespace and, in the end, what is a namespace" left unanswered. Two days ago I had the opportunity to borrow this book from a friend of mine, old-time programmer and IT PM, and I couldn't left it out of my hands. So, first advice, start with this book if you want to understand basics of XML. It starts from the scratch, but assuming you have some knowledge about markup, web technologies and, especially, some more insight into HTML and text tagging, but, otherwise, the content is so well written and refined that you will not have any problem in understanding the concepts. You need a good XML editor. I downloaded XMLSpy trial version from Alltova but I think any good XML editor will do. You will need it to write down and test the collection of examples that the book is filled with. After you got your XML editor,

you may start reading the book, "pencil in hand" to try all the examples provided. By the way, I reached page 183 (out of 1000+) and no typos encountered yet. The code within the examples is very well written, no mistakes up to now. Since I read only a quarter of the information, I will stick to the chapters I went through so far.

[Download to continue reading...](#)

Beginning XML, 4th Edition XML & XSL Fast Start 2nd Edition: Your Quick Start Guide for XML & XSL XML Programming Success in a Day: Beginner's Guide to Fast, Easy, and Efficient Learning of XML Programming A Designer's Guide to Adobe InDesign and XML: Harness the Power of XML to Automate your Print and Web Workflows Securing Web Services with WS-Security: Demystifying WS-Security, WS-Policy, SAML, XML Signature, and XML Encryption The Xml Files: Development of Xml/Xsl Applications Using Websphere Studio Version 5 (Ibm Redbooks) The XML Files : Using XML and XSL with IBM WebSphere V3.0 (IBM Redbook) Beginning XML Beginning Java Databases: JDBC, SQL, J2EE, EJB, JSP, XML The Beginning Band Fun Book's FUNsembles: Book of Easy Duets (French Horn): for Beginning Band Students The Beginning Band Fun Book's FUNsembles: Book of Easy Duets (Trombone): for Beginning Band Students Beginning Tennis (Beginning Sports) Beginning JavaScript with DOM Scripting and Ajax: From Novice to Professional (Beginning: From Novice to Professional) Beginning HTML with CSS and XHTML: Modern Guide and Reference (Beginning: from Novice to Professional) Beginning Web Programming with HTML, XHTML, and CSS (Wrox Beginning Guides) Beginning Perl Web Development: From Novice to Professional (Beginning: From Novice to Professional) Sixty: A Diary of My Sixty-First Year: The Beginning of the End, or the End of the Beginning? Beginning World Music for Piano: Beginning Piano Series The Beginning Band Fun Book's FUNsembles: Book of Easy Duets (Alto Saxophone): for Beginning Band Students Special Edition Using Java 2 Enterprise Edition (J2EE): With JSP, Servlets, EJB 2.0, JNDI, JMS, JDBC, CORBA, XML and RMI

[Dmca](#)