Beginning XML
4th Edition

David Hunter,
Jeff Rafter,
Joe Fawcett,
Eric van der Vlist,
Danny Ayers,
Jon Duckett,
Andrew Watt, and
Linda McKinnon

BICENTENNIAL 1807 WILEY 2007 BICENTENNIAL
Wiley Publishing, Inc.

Beginning XML

4th Edition

Beginning XML
4th Edition

David Hunter,
Jeff Rafter,
Joe Fawcett,
Eric van der Vlist,
Danny Ayers,
Jon Duckett,
Andrew Watt, and
Linda McKinnon

BICENTENNIAL 1807 WILEY 2007 BICENTENNIAL

Wiley Publishing, Inc.

Beginning XML, 4th Edition

Published by
Wiley Publishing, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

Copyright © 2007 by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-0-470-11487-2

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data:

Beginning XML / David Hunter ... [et al.]. -- 4th ed.
p. cm.
ISBN 978-0-470-11487-2 (paper/website)
1. XML (Document markup language) I. Hunter, David, 1974 May 7-
QA76.76.H94B439 2007

006.7'4--dc22
2007006580

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Legal Department, Wiley Publishing, Inc., 10475 Crosspoint Blvd., Indianapolis, IN 46256, (317) 572-3447, fax (317) 572-4355, or online at www.wiley.com/go/permissions.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Trademarks: Wiley, the Wiley logo, Wrox, the Wrox logo, Programmer to Programmer, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc., is not associated with any product or vendor mentioned in this book.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

I would like to thank God, for continuing to give me opportunities to do
what I love; my church family, for giving me more support than I deserve;
and Andrea, for giving me more support than anyone deserves.
I would also like to thank the editors, for their constant help.
Their dedication to the quality of this book was a major factor in its success.
—David

To Ali and Jude, for their loving patience.
—Jeff

To my two brothers, Peter and Stephen, who have both helped me in my life
and career in their own ways, many thanks.
—Joe

To my wife, Catherine, and children, Deborah, David, Samuel, and Sarah,
for their patience and support while I am busy writing books.
—Eric

To my late grandmother, Mona Cartledge, who once gave me a
Commodore Pet.
—Danny

About the Authors

David Hunter is a Senior Technical Consultant for CGI, a full-service IT and business process services partner. Providing technical leadership and guidance for solving his clients' business problems, he is a jack-of-all-trades and master of some. With a career that has included design, development, support, training, writing, and other roles, he has had extensive experience building scalable, reliable, enterprise-class applications. David loves to peek under the hood at any new technology that comes his way, and when one catches his fancy, he really gets his hands dirty. He loves nothing more than sharing these technologies with others.

Jeff Rafter is an independent consultant based in Redlands, California. His focus is on emerging technology and web standards, including XML and validation. He currently works with Baobab Health Partnership with a focus on improving world health.

Joe Fawcett (`http://joe.fawcett.name`) started programming in the 1970s and worked briefly in IT when leaving full-time education. He then pursued a more checkered career before returning to software development in 1994. In 2003 he was awarded the title of Microsoft Most Valuable Professional in XML for community contributions and technical expertise; he has subsequently been re-awarded every year since. Joe currently works in London and is head of software development for FTC Kaplan Ltd., a leading international provider of accountancy and business training.

Eric van der Vlist is an independent consultant and trainer. His domains of expertise include web development and XML technologies. He is the creator and main editor of XMLfr.org, the main site dedicated to XML technologies in French, the lead author of *Professional Web 2.0 Programming*, the author of the O'Reilly animal books *XML Schema* and *RELAX NG* and a member or the ISO DSDL (`http://dsdl.org`) working group focused on XML schema languages. He is based in Paris and can be reached at `vdv@dyomedea.com`, or meet him at one of the many conferences where he presents his projects.

Danny Ayers is a freelance developer and consultant specializing in cutting-edge web technologies. His blog (`http://dannyayers.com`) tends to feature material relating to the Semantic Web and/or cat photos.

Linda McKinnon has more than 10 years of experience as a successful trainer and network engineer, assisting both private and public enterprises in network architecture design, implementation, system administration, and RFP procurement. She is a renowned mentor and has published numerous Linux study guides for Wiley Press and Gearhead Press.

Credits

Senior Acquisitions Editor
Jim Minatel

Development Editors
Sara Shlaer
Lisa Thibault

Technical Editor
Phred Menyhert

Production Editor
William A. Barton

Copy Editor
Luann Rouff

Editorial Manager
Mary Beth Wakefield

Production Manager
Tim Tate

Vice President and Executive Group Publisher
Richard Swadley

Vice President and Executive Publisher
Joseph B. Wikert

Graphics and Production Specialists
Brooke Graczyk
Denny Hager
Joyce Haughey
Jennifer Mayberry
Barbara Moore
Alicia B. South

Quality Control Technician
John Greenough

Project Coordinator
Lynsey Osborn

Media Development Specialists
Angie Denny
Kit Malone
Kate Jenkins
Steve Kudirkan

Proofreading
Aptara

Indexing
Broccoli Information Management

Anniversary Logo Design
Richard Pacifico

Acknowledgments

This book would not have been possible without the work of the many developers dedicated to improving the Web through standards. We would also like to thank the countless contributors to mailing lists, IRC channels, forums, and friends that have helped us through the difficult corners of the specifications and technologies presented in this book.

Thanks to Nicholas C. Zakas for his ideas and assistance in implementing the AutoSuggest Control. Many thanks to Phillip Pearson, who runs TopicExchange.com. He provided much-needed technical support that otherwise would have meant rewriting most of Chapter 14. We would also like to thank Jim Ley and Doug Schepers for their assistance on the case study and Chapter 19. Special thanks to our lead editor, Sara Shlaer, for her gentle and not so gentle persuasive powers and attention to detail; to editor Lisa Thibault, for her thoughtful assistance; and to Phred Menyhert, for a rigorous technical edit. Many thanks to our acquisitions editor, Jim Minatel, who has shepherded this book through many incarnations.

Contents

Acknowledgments **ix**
Introduction **xxvii**

Part I: Introduction 1

Chapter 1: What Is XML? 3

Of Data, Files, and Text **3**
Binary Files 4
Text Files 5
A Brief History of Markup 6
So What Is XML? **7**
What Does XML Buy Us? 10
HTML and XML: Apples and Red Delicious Apples 13
Hierarchies of Information 14
What's a Document Type? 17
No, Really — What's a Document Type? 18
Origin of the XML Standards **18**
What Is the World Wide Web Consortium? 18
Components of XML 19
Where XML Can Be Used, and What You Can Use It For **20**
Reducing Server Load 20
Website Content 20
Distributed Computing 21
e-Commerce 21
Summary **22**
Exercise Questions **22**
Question 1 22
Question 2 22

Chapter 2: Well-Formed XML 23

Parsing XML **24**
Tags and Text and Elements, Oh My! **24**
Rules for Elements 31

Attributes **39**
When to Use Attributes 43
Comments **45**
Empty Elements **49**
XML Declarations **50**
Version 51
Encoding 51
Standalone 53
Processing Instructions **56**
Illegal PCDATA Characters **59**
Escaping Characters 60
CDATA Sections 61
Errors in XML **64**
Summary **64**
Exercise Questions **65**
Question 1 65
Question 2 65

Chapter 3: XML Namespaces **67**

Why We Need Namespaces **67**
Using Prefixes 69
Why Doesn't XML Just Use These Prefixes? 70
How XML Namespaces Work **72**
Default Namespaces 75
Do Different Notations Make Any Difference? 81
Namespaces and Attributes 83
Understanding URIs **86**
URLs 86
URNs 87
Why Use URLs for Namespaces, Not URNs? 87
What Do Namespace URIs Really Mean? 88
RDDL 89
When to Use Namespaces **89**
Summary **90**
Exercise Questions **91**
Question 1 91
Question 2 91
Question 3 91

Part II: Validation 93

Chapter 4: Document Type Definitions 95

Running the Samples 96
Preparing the Ground 96
The Document Type Declaration 100
Sharing Vocabularies 104
Anatomy of a DTD 105
Element Declarations 105
Attribute Declarations 120
Entities 131
Developing DTDs 141
DTD Limitations 142
DTD Syntax 142
XML Namespaces 143
Data Typing 143
Limited Content Model Descriptions 143
Summary 143
Exercise Questions 144
Question 1 144
Question 2 144
Question 3 144

Chapter 5: XML Schemas 145

Benefits of XML Schemas 146
XML Schemas Use XML Syntax 146
XML Schema Namespace Support 146
XML Schema Data Types 147
XML Schema Content Models 147
Do We Still Need DTDs? 147
XML Schemas 148
The XML Schema Document 148
Running the Samples 148
<schema> Declarations 152
<element> Declarations 155
<complexType> Declarations 165
<group> Declarations 167
Content Models 168
<attribute> Declarations 177
<attributeGroup> Declarations 183

Creating Elements with Simple Content and Attributes 185
Datatypes 186
<simpleType> Declarations 193
Creating a Schema from Multiple Documents 200
<import> Declarations 200
<include> Declarations 204
Documenting XML Schemas 206
Comments 206
Attributes from Other Namespaces 207
Annotations 208
Summary 209
Exercise Questions 210
Question 1 210
Question 2 210
Question 3 210

Chapter 6: RELAX NG 211

XML and Compact Syntaxes 212
Running the Samples 212
RELAX NG Patterns 213
Element, Attribute, and Text Patterns 213
Combining and Reusing Patterns and Grammars 227
Named Patterns 227
Combining Named Pattern Definitions 230
Schema Modularization Using the include Directive 231
Redefining Included Named Patterns 232
Removing Patterns with the notAllowed Pattern 233
Extensions and Restrictions 234
Nested Grammars 235
Additional RELAX NG Features 236
Namespaces 236
Name-Classes 237
Datatypes 241
List Patterns 243
Comments and Divisions 244
Useful Resources 245
Summary 245
Exercise Questions 245
Question 1 246
Question 2 246

Part III: Processing 247

Chapter 7: XPath 249

Ways of Looking at an XML Document 250
Modeling XML Documents 250
Visualizing XPath 251
Understanding Context 252
What Is a Node? 254
XPath 1.0 Types 257
Abbreviated and Unabbreviated Syntax 259
XPath 1.0 Axes 260
Child Axis 260
attribute Axis 262
ancestor Axis 264
ancestor-or-self Axis 265
descendant Axis 265
descendant-or-self Axis 266
following Axis 266
following-sibling Axis 268
namespace Axis 268
parent Axis 271
preceding Axis 271
preceding-sibling Axis 272
self Axis 273
XPath 1.0 Functions 274
Boolean Functions 274
Node-Set Functions 275
Numeric Functions 275
String Functions 276
Predicates 278
Structure of XPath Expressions 278
XPath 2.0 281
Revised XPath Data Model 281
W3C XML Schema Data Types 281
Additional XPath 2.0 Functions 282
XPath 2.0 Features 282
Summary 285
Exercise Questions 285
Question 1 286
Question 2 286

Chapter 8: XSLT 287

What Is XSLT? 287
Restructuring XML 288
Presenting XML Content 288
How an XSLT Processor Works 288
Running the Examples 289
Introducing the Saxon XSLT Processor 289
Installing the Saxon XSLT Processor 290
Procedural versus Declarative Programming 292
Procedural Programming 292
Declarative Programming 292
Foundational XSLT Elements 293
The <xsl:stylesheet> Element 295
The <xsl:template> Element 296
The <xsl:apply-templates> Element 296
Getting Information from the Source Tree 297
The <xsl:value-of> Element 297
The <xsl:copy> Element 299
The <xsl:copy-of> Element 303
Influencing the Output with the <xsl:output> Element 306
Conditional Processing 306
The <xsl:if> Element 306
The <xsl:choose> Element 308
The <xsl:for-each> Element 311
The <xsl:sort> Element 312
XSLT Modes 314
XSLT Variables and Parameters 320
Named Templates and the <xsl:call-template> Element 322
XSLT Functions 323
XSLT 2.0 323
Grouping in Version 2.0 324
Non-XML Input and String Handling 327
Multiple Outputs 330
User-Defined Functions 332
xsl:value-of changes 334
Summary 335
Exercise Questions 335
Question 1 335
Question 2 335

Part IV: Databases 337

Chapter 9: XQuery, the XML Query Language 339

Why XQuery? 340
Historical Factors 340
Technical Factors 340
Current Status 341
XQuery Tools 343
Saxon 343
X-Hive.com Online 345
X-Hive Database 346
Tamino Database 346
Microsoft SQL Server 2005 346
Oracle 346
Some XQuery Examples 346
Input Functions 346
Retrieving Nodes 348
Element Constructors 351
The XQuery Prolog 355
Computed Constructors 358
Syntax 359
The XQuery Data Model 360
Shared Data Model with XPath 2.0 and XSLT 2.0 360
Node Kinds 361
Sequences of Node-Sets 361
Document Order 361
Comparing Items and Nodes 361
Types in XQuery 361
Axes in XQuery 361
XQuery Expressions 362
FLWOR Expressions 362
XQuery Functions 368
The concat() Function 369
The count() Function 369
Using Parameters with XQuery 370
User-Defined Functions 371
Looking Ahead 372
Update Functionality 372
Full-Text Search 372
Summary 372

Exercise Questions **373**
Question 1 373
Question 2 373

Chapter 10: XML and Databases **375**

The Need for Efficient XML Data Stores **375**
The Increasing Amount of XML 376
Comparing XML-Based Data and Relational Data 377
Approaches to Storing XML **378**
Storing XML on File Systems 378
Using XML With Conventional Databases 379
Native XML Databases 381
Using Native XML Databases **382**
Obtaining and Installing eXist 382
Interacting with eXist 384
XML in Commercial RDBMSs **395**
XML Functionality in SQL Server 2000 395
Web Service Support 426
XML in Open Source RDBMS **426**
Installing MySQL 426
Adding Information in MySQL 427
Querying MySQL 430
Updating XML in MySQL 435
Usability of XML in MySQL 436
Client-Side XML Support 437
Choosing a Database to Store XML **438**
Looking Ahead **438**
Summary **438**
Exercise Questions **438**
Question 1 439
Question 2 439
Question 3 439

Part V: Programming **441**

Chapter 11: The XML Document Object Model (DOM) **443**

Purpose of the XML DOM **443**
Interfaces and Objects 445
The Document Object Model at the W3C **446**
XML DOM Implementations 447

Two Ways to View DOM Nodes **448**
Overview of the XML DOM 448
Tools to Run the Examples **450**
Browser Differences 450
The Node Object **458**
Properties of the Node Object 458
Methods of the Node Object 461
Loading an XML Document 462
The Effect of Text Nodes 468
The NamedNodeMap Object 471
The NodeList Object 475
The DOMException Object 476
The Document Interface **478**
How the XML DOM Is Used in InfoPath 2007 **481**
Summary **482**
Exercise Questions **482**
Question 1 482
Question 2 482

Chapter 12: Simple API for XML (SAX) **483**

What Is SAX and Why Was It Invented? **483**
A Brief History of SAX 484
Where to Get SAX 485
Setting Up SAX 486
Receiving SAX Events **486**
ContentHandler Interface 487
ErrorHandler Interface 504
DTDHandler Interface 509
EntityResolver Interface 510
Features and Properties 510
Extension Interfaces 514
Good SAX and Bad SAX **515**
Consumers, Producers, and Filters **516**
Other Languages **516**
Summary **517**
Exercise Questions **518**
Question 1 518
Question 2 518

Part VI: Communication 519

Chapter 13: RSS, Atom, and Content Syndication 521

Syndication and Meta Data 521
Syndication Systems 522
The Origin of RSS Species 525
RSS-DEV and RSS 1.0 529
UserLand and RSS 2.0 531
Atom 533
Working with News Feeds 536
Newsreaders 536
Data Quality 536
A Simple Aggregator 537
Modeling Feeds 537
Program Flow 540
Implementation 540
Transforming RSS with XSLT 557
Useful Resources 567
Summary 568
Exercise Questions 568
Question 1 569
Question 2 569

Chapter 14: Web Services 571

What Is an RPC? 571
RPC Protocols 573
DCOM 573
IIOP 574
Java RMI 575
The New RPC Protocol: Web Services 575
XML-RPC 576
The Network Transport 579
Taking a REST 596
The Web Services Stack 600
SOAP 600
WSDL 601
UDDI 602
Surrounding Specifications 602
Summary 604

Exercise Questions **605**
Question 1 605
Question 2 605

Chapter 15: SOAP and WSDL **607**

Laying the Groundwork **608**
Running Examples in Windows 2003, XP, and 2000 **608**
The New RPC Protocol: SOAP **608**
Just RESTing 612
Basic SOAP Messages 613
More Complex SOAP Interactions 620
Defining Web Services: WSDL **632**
<definitions> 633
<types> 633
<messages> 634
<portTypes> 635
<binding> 635
<soap:body> 637
<service> 638
Other Bindings 641
Summary **644**
Exercise Questions **644**
Question 1 644
Question 2 644

Chapter 16: Ajax **645**

Early Attempts at Asynchronous Updates **645**
Microsoft versus Mozilla 647
Cross-Browser Solutions 647
Basic Posting Techniques **649**
Transport and Processing on the Server **652**
JSON 652
Payment Card Validator 653
The AutoSuggest Box 658
Server-Side Proxies **681**
The Currency Converter Proxy 682
Summary **686**
Exercise Questions **687**
Question 1 687
Question 2 687
Question 3 687

Part VII: Display 689

Chapter 17: Cascading Style Sheets (CSS) 691

Why Stylesheets? 692
Introducing CSS 693
CSS Properties 694
Inheritance 695
Attaching the Stylesheet to an XML Document 699
Selectors 700
Using CSS for Layout of XML Documents 701
Understanding the Box Model 702
Positioning in CSS 706
Laying Out Tabular Data 719
Links in XML Documents 721
XLink Support in Firefox 721
Forcing Links Using the XHTML Namespace 725
Images in XML Documents 725
Using CSS to Add Content to Documents 726
Attribute Content 729
Attribute Selectors 729
Using Attribute Values in Documents 729
Summary 732
Exercise Questions 732
Question 1 732
Question 2 733
Question 3 733
Question 4 733

Chapter 18: XHTML 735

Separating Style from Content 736
Learning XHTML 1.x 738
Document Type Definitions for XHTML 738
Basic Changes in Writing XHTML 740
Styling XHTML Documents 751
Stricter Documents Make Faster and Lighter Processors 753
XHTML Tools 753
Validating XHTML Documents 754
Validation Pitfalls 756
Mime Types Pitfalls 757

Modularized XHTML **759**
Module Implementations 761
XHTML 1.1 761
XHTML Basic 762
What's Next for XHTML **763**
Summary **765**
Exercise Questions **766**
Question 1 766
Question 2 766

Chapter 19: Scalable Vector Graphics (SVG) **767**

What Is SVG? **767**
Scalable, Vector, Graphics 768
Putting SVG to Work 769
An SVG Toolkit 769
Getting Started **771**
Views and Units 774
The Painter's Model 774
Grouping 776
Transformations 776
Paths 777
Images 780
Text 781
Comments, Annotation, and Metadata 782
Scripting 784
SVG on Your Website 785
Tangram: A Simple Application **786**
XHTML Wrapper 787
SVG Shapes 788
Tangram Script 792
Useful Resources **799**
Summary **800**
Exercise Questions **800**
Question 1 800
Question 2 801

Chapter 20: XForms **803**

How XForms Improves on HTML Forms **804**
XForms Tools **804**
An Illustrative XForms Example **810**

XForms Form Controls **817**
The xforms:input Element 817
The xforms:secret Element 818
The xforms:textarea Element 818
The xforms:output Element 818
The xforms:upload Element 819
The xforms:range Element 819
The xforms:trigger Element 820
The xforms:submit Element 820
The xforms:select Element 821
The xforms:select1 Element 822
Constraining XForms Instances **828**
The xforms:bind Element 828
W3C XML Schema in XForms 833
Schema or Bind Elements: Which One to Choose? 834
XForms Events **834**
The XForms Action Module **835**
Developing and Debugging XForms 836
Alternatives to XForms **836**
Microsoft InfoPath 836
Adobe LiveCycle 837
HTML Forms 838
Summary **838**
Exercise Questions **838**
Question 1 838
Question 2 838

Part VIII: Case Study **839**

Chapter 21: Case Study: Payment Calculator **841**

Mortgage Calculations **841**
What You'll Need **842**
Online Loan Calculator **842**
Integrating the Calculation Web Service 849
Enhancing the Display with SVG 865
Adding the Frame to the Main Page 868
Summary **872**

Chapter 22: Case Study: Payment Calculator — Ruby on Rails **Online**

Appendix A: Exercise Solutions 873

Appendix B: XPath Reference 923

Appendix C: XSLT Reference 939

Appendix D: The XML Document Obect Model Online

Appendix E: XML Schema Element and Attribute Reference Online

Appendix F: XML Schema Datatypes Reference Online

Appendix G: SAX 2.0.2 Reference Online

Index 971

Introduction

Welcome to Beginning XML, Fourth Edition, the book I wish I'd had when I was first learning the language!

When we wrote the first edition of this book, 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 we started 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. As we began 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. 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 access—for which some of the technology is XML-based). However, there are many other non-web-based applications for which XML is useful—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 XML—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?

Who Is This Book For?

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.

How This Book Is Organized

We've arranged the subjects covered in this book to take you from novice to expert in as logical a manner as we could. In this Fourth Edition, we have structured the book in sections that are 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.

- We begin 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 we show 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, we unleash the programmer within and look 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.