

CHARLES F. GOLDFARB'S

# XML

HANDBOOK™

4th EDITION

FREE  
Trial Version  
turboXML™ IDE & Schema Editor



**FREE**

NeoCore™ XMS

Native XML Database—Personal Edition

- ▶ The proven XML resource: applications, products, technologies, and **tutorials!**
- ▶ Revised and enlarged—latest standards and trends: **schemas**, datatypes, XSL, voice, wireless
- ▶ Two CD-ROMs: 175 genuinely free software packages, including the **IBM alphaWorks** suite
- ▶ **Web services:** SOAP, WSDL, UDDI

100,000  
copies in  
print

CHARLES F. GOLDFARB • PAUL PRESCOD

THE CHARLES F. GOLDFARB DEFINITIVE XML SERIES™

ADOBE® FRAMEMAKER®+SGML—FREE TRYOUT

# **Charles F. Goldfarb's XML Handbook™**

## **Fourth Edition**

■ Charles F. Goldfarb

■ Paul Prescod

江苏工业学院图书馆  
藏书章



Prentice Hall PTR, Upper Saddle River, NJ 07458  
[www.phptr.com](http://www.phptr.com)

Library of Congress Cataloging-in-Publication Data

Goldfarb, Charles F.

Charles F. Goldfarb's XML Handbook / Charles F. Goldfarb, Paul Prescod.—4th ed.

p. cm.—(The Charles F. Goldfarb Definitive XML Series)

ISBN 0-13-065198-2

I. XML (Document markup language). I. Prescod, Paul. II. Series.

QA76.76.H94 G65 2001b

005.72—dc21

Editorial/Production Supervisor: *Faye Gemmellaro*

Acquisitions Editor: *Mark L. Taub*

Editorial Assistant: *Sarah Hand*

Marketing Manager: *Bryan Gambrel*

Manufacturing Manager: *Maura Zaldivar*

Cover Designer: *Anthony Gemmellaro*

Cover Design Director: *Jerry Votta*

Series Designer: *Gail Cocker-Bogusz*



© 2002 Charles F. Goldfarb

Published by Prentice Hall PTR

Prentice-Hall, Inc.

Upper Saddle River, NJ 07458

Prentice Hall books are widely used by corporations and government agencies for training, marketing, and resale.

The publisher offers discounts on this book when ordered in bulk quantities. For more information, contact: Corporate Sales Department, Phone: 800-382-3419; Fax: 201-236-7141; E-mail: [corpsales@prenhall.com](mailto:corpsales@prenhall.com); or write: Prentice Hall PTR, Corp. Sales Dept., One Lake Street, Upper Saddle River, NJ 07458.

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher.

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-065198-2

Pearson Education Ltd.

Pearson Education Australia Pty. Limited

Pearson Education Singapore, Pte.Ltd.

Pearson Education North Asia Ltd.

Pearson Education Canada, Ltd.

Pearson Educación de Mexico, S.A. de C.V.

Pearson Education—Japan

Pearson Education Malaysia, Pte. Ltd.

Adobe, the Adobe logo, Acrobat, FrameMaker, and PostScript are trademarks of Adobe Systems Incorporated. Microsoft and Windows are registered trademarks of Microsoft Corporation in the U.S. and other countries. Oracle is a registered trademark, and Oracle8i is a trademark of Oracle Corporation. IBM and alphaWorks are registered trademarks of IBM Corporation in the U.S. and other countries. Intel and the Intel logo are registered trademarks and Intel NetStructure and VTune are trademarks of Intel Corporation.

Sun, Sun Microsystems, Java and all Java-based marks, Enterprise JavaBeans, JDBC, JavaServer Pages, and the Sun Logo are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. The authors and publisher of this book are independent of Sun Microsystems.

CSS, DOM, MathML, RDF, SMIL, SVG, W3C, WAI, XHTML, XML, and XSL are trademarks (one or more of which are registered in numerous countries) of the World Wide Web Consortium; marks of W3C are registered and held by its host institutions MIT, INRIA, and Keio.

The XML Handbook, Charles F. Goldfarb's XML Handbook, XML Handbook Jargon Demystifier, and HARP are trademarks of Charles F. Goldfarb.

Names such as company names, trade names, font names, service names, and product names appearing in this book may be registered or unregistered trademarks or service marks, whether or not identified as such. All such names and all registered and unregistered trademarks, service marks, and logos appearing in this book or on its cover are used for identification purposes only and are the property of their respective owners.

Series logo by Dmitry Kirsanov and Charles F. Goldfarb, copyright ©2002 Charles F. Goldfarb.

Excerpts from International Standards are copyrighted by the International Organization for Standardization, and are included with the kind permission of the copyright owner. Complete copies of International Standards can be obtained from the national member body of ISO in your country, or contact ISO, case postale 56, CH-1211 Geneva 20, Switzerland.

Excerpts from World Wide Web Consortium documents are included in accordance with the W3C IPR Document Notice, <http://www.w3.org/Consortium/Legal/copyright-documents.html>. Copyright © World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

The development of this book was partly subsidized by Sponsors, who provided both financial support and expert assistance in preparing the initial draft of the text identified with their names and in reviewing the edited version at varying stages of completion. However, as the Authors exercised final editorial control over the book, the Sponsors are not responsible for errors that may have been introduced during that process. Opinions expressed in this book are those of the Authors and are not necessarily those of the Series Editor, Sponsors, or Publisher.

This book, and the CD-ROM included with it, contain software and descriptive materials provided by (or adapted from materials made publicly available by) product developers, vendors, and service providers. Said software and materials have not been reviewed, edited, or tested, and neither the Authors, Contributors, Series Editor, Publisher, Sponsors, or other parties connected with this book are responsible for their accuracy or reliability. Readers are warned that they use said software and materials at their own risk, and are urged to test the software and confirm the validity of the information prior to use.

This book contains forward-looking statements about publicly-traded companies. These forward-looking statements are made pursuant to the safe harbor provisions of the U.S. Private Securities Litigation Reform Act of 1995. Investors are cautioned that all forward-looking statements

involve risk and uncertainties including, without limitation: risks of intellectual property litigation; risks in technology development and commercialization; risks in product development and market acceptance of and demand for the company's products; risks of downturns in economic conditions generally and in the markets for Web, Internet, and intranet software and services markets specifically; risks associated with competition and competitive pricing pressures; risks associated with foreign sales and higher customer concentration; and other risks detailed in each company's filings with the Securities and Exchange Commission.



# Preface

When Paul Prescod and I wrote the first edition of this book – four years and 100,000 copies ago – XML was brand new and the subject of extraordinary hype. It promised to provide universal data interchange, revolutionize publishing on the Web, and transform distributed computing.

Those claims were amazing, not just because of the extent of the promised impact, but because of the diversity of the areas affected. More amazingly, the claims have largely been fulfilled. With the support of the entire computer industry, an XML-based infrastructure is being constructed for modern computing; indeed, for modern business itself.

In some ways, though, the construction site resembles the Tower of Babel. The professionals in the areas affected by XML tend to talk and write about it in their own way, from each area's unique perspective, and in its specialized jargon.

But not in *The XML Handbook*!

From the first edition, our aim has been to integrate and unify the teaching of XML so that any tech industry professional can learn it, regardless of background. And by "learn it" we mean not just the technical details but the way that XML is used. Specifically:

- We use a unified standards-based vocabulary consistently. We explain when particular disciplines or industries use terms in conflicting or ambiguous ways.
- We explain all technical concepts as we introduce them, even the basics, but we don't indulge in "simplification by distortion". We clarify without sacrificing accuracy.
- We describe major trends, applications, and product categories objectively, employing the unified vocabulary, so you can see clearly how they relate to one another and to XML technology.

As a result, developers with diverse backgrounds found they could get the full picture of XML from *The XML Handbook*. Moreover, they also found they could encourage management to read the book and learn why XML is so important to the enterprise.

### *XML in a nutshell*

HTML – the HyperText Markup Language – made the Web the world's library. XML – the Extensible Markup Language – is its sibling, and it is making the Web the world's commercial and financial hub.

In the process, the Web is becoming much more than a static library. Increasingly, users are accessing the Web for "Web pages" that aren't actually on the shelves. Instead, the pages are generated dynamically from information available to the Web server. That information can come from databases on the Web server, from the site owner's enterprise databases, or even from other websites.

And that dynamic information needn't be served up raw. It can be analyzed, extracted, sorted, styled, and customized to create a personalized Web experience for the end-user. To coin a phrase, Web *pages* are evolving into Web *services*.

For this kind of power and flexibility, XML is the markup language of choice. You can see why by comparing XML and HTML. Both are based on SGML – the International Standard for structured information – but look at the difference:

In HTML:

```
<p>P200 Laptop  
<br>Friendly Computer Shop  
<br>$1438
```

In XML:

```
<product>  
<model>P200 Laptop</model>  
<dealer>Friendly Computer Shop</dealer>  
<price>$1438</price>  
</product>
```

Both of these may appear the same in your browser, but the XML data is *smart* data. HTML tells how the data should *look*, but XML tells you what it *means*.

With XML, your browser knows there is a product, and it knows the model, dealer, and price. From a group of these it can show you the cheapest product or closest dealer without going back to the server.

Unlike HTML, XML allows custom tags that can describe exactly what you need to know. Because of that, your client-side applications can access data sources anywhere on the Web, in any format. New “middle-tier” servers sit between the data sources and the client, translating everything into your own task-specific XML.

But XML data isn't just smart data, it's also a smart document. That means when you display the information, the model name can be in a different font from the dealer name, and the lowest price can be highlighted in green. Unlike HTML, where text is just text to be rendered in a uniform way, with XML text is smart, so it can control the rendition.

And you don't have to decide whether your information is data or documents; in XML, it is always both at once. You can do data processing or document processing or both at the same time.

With that kind of flexibility, it's no wonder that we're starting to see a new Web of smart, structured information. It's a “Semantic Web” in which computers understand the meaning of the data they share.

Your broker sends your account data to Quicken using XML. Your imaging software keeps its templates in XML. Everything from math to multimedia, chemistry to commerce, wireless to Web services, is using XML or is preparing to start.

*The XML Handbook* will help you get started too!



*What about SGML?*

This book is about XML. You won't find feature comparisons to SGML, or footnotes with nerdy observations like "the XML empty-element tag does not contradict the rule that every element has a start-tag and an end-tag because, in SGML terms, it is actually a start-tag followed immediately by a null end-tag".<sup>1</sup>

Nevertheless, for readers who use SGML, it is worth addressing the question of how XML and SGML relate. There has been a lot of speculation about this.

Some claim that XML will replace SGML because there will be so much free and low-cost software. Others assert that XML users, like HTML users before them, will discover that they need more of SGML and will eventually migrate to the full standard.

The truth is that XML is a simplified subset of SGML. The subsetting was optimized for the Web environment, which implies data-processing-oriented (rather than publishing-oriented), short life-span (in fact, usually dynamically-generated) information. The vast majority of XML documents will be created by computer programs and processed by other programs, then destroyed. Humans will never see them.

Eliot Kimber, who was a member of both the XML and SGML standards committees, says:

There are certain use domains for which XML is simply not sufficient and where you need the additional features of SGML. These applications tend to be very large scale and of long term; e.g., aircraft maintenance information, government regulations, power plant documentation, etc.

Any one of them might involve a larger volume of information than the entire use of XML on the Web. A single model of commercial aircraft, for example, requires some four million unique pages of documentation that must be revised and republished quarterly. Multiply that by the number of models produced by companies like Airbus and Boeing and you get a feel for the scale involved.

---

1. Well, yes, I did just make that nerdy observation, but it wasn't a footnote, was it?

I agree with Eliot. I invented SGML, I'm proud of it, and I'm awed that such a staggering volume of the world's mission-critical information is represented in it.

I'm gratified that SGML made the Web possible and that the Society for Technical Communication awarded joint Honorary Fellowships to the Web's inventor, Tim Berners-Lee, and myself in recognition of the synergy.

But I'm also proud of XML. I'm proud of my friend Jon Bosak who made it happen, and I'm glad that the World Wide Web is becoming XML-based.

If you are new to XML, don't worry about any of this. All you need to know is that the XML subset of SGML has been in use for a decade or more, so you can trust it.

SGML still keeps the airplanes flying, the nuclear plants operating safely, and the defense departments in a state of readiness. You should look into it if you produce documents on the scale of an Airbus or Boeing. For the rest of us, there's XML.

### *About our sponsors*

With all the buzz surrounding a hot technology like XML, it can be tough for a newcomer to distinguish the solid projects and realistic applications from the fluff and the fantasies. It is tough for authors as well, to keep track of all that is happening in a field expanding as rapidly as this one.

In this case, the solution to both problems was to seek support and expert help from friends in the industry. I know the leading companies in the XML arena and know they have experience with both proven and leading-edge applications and products.

In the usual way of doing things, had we years to write this book, we would have interviewed each company to learn about its strategies, products and/or application experiences, written some chapters, asked the companies to review them, etc., and gone on to the next company. To save time and improve accuracy, we engaged in parallel processing. I spoke with each sponsor, agreed on subject matter for a chapter that would fit the book plan, and asked them to write the first draft.

I used their materials as though they were my own interview notes – editing, rewriting, deleting, and augmenting as necessary to achieve my objective for the chapter in the context of the book. I used consistent standards-based terminology and an objective factual style. All sponsored chapters are identified with the name of the sponsor, and usually with the

names of the experts who contributed to it. I'd like to take this opportunity to thank them for being so generous with their time and knowledge.

We are grateful to our sponsors just as we are grateful to you, our readers. Both of you together make it possible for *The XML Handbook* to exist. In the interests of everyone, we make our own editorial decisions and we don't recommend or endorse any product or service offerings over any others.

Our thirty-four sponsors are:

- Adobe Systems Incorporated,  
<http://www.adobe.com/products/framesmaker>
- Altova, Inc., <http://www.xmlspy.com>
- Auto-trol Technology, <http://www.auto-trol.com>
- Business Layers, <http://www.businesslayers.com>
- Chrystal Software, <http://www.chrystal.com>
- eidon GmbH, <http://www.eidon-products.com>
- Four J's Development Tools, <http://www.4js.com>
- Hewlett-Packard Company, <http://www.hp.bluestone.com>
- HostBridge Technology, <http://www.hostbridge.com>
- IBM Corporation, <http://www.ibm.com/xml>
- IBM Informix, <http://www.informix.com>
- ICE Authoring Group, <http://www.icestandard.org>
- Infoteria Corporation, <http://www.infoteria.com>
- Innodata Corporation, <http://www.innodata.com>
- Intel Corporation, <http://www.intel.com/eBusiness>
- Intelligent Compression Technologies, Inc.,  
<http://www.ictcompress.com>
- Interwoven, Inc., <http://www.interwoven.com>
- Liquent, <http://www.liquent.com>
- Microsoft Corporation, <http://msdn.microsoft.com/xml>
- Mozquito Technologies AG, <http://www.mozquito.com>
- NeoCore, <http://www.neocore.com>
- Oracle Corporation, <http://www.oracle.com/xml>
- Peregrine Systems, <http://www.peregrine.com>
- Profium Ltd., <http://www.profium.com>
- Sequoia Software, <http://www.sequoiasoftware.com>
- Software AG, <http://www.softwareagusa.com>
- Sun Microsystems, <http://java.sun.com/xml>
- Synth-Bank LLC, <http://www.synthbank.com>
- TIBCO Software Inc., <http://www.extensibility.com>

- Vertical Computer Systems/Emily Solutions,  
<http://www.emilysolutions.com>
- Voxeo Corporation, <http://community.voxeo.com>
- XCare.net, <http://www.xcare.net>
- XMLCities, Inc., <http://www.XMLCities.com>
- XyEnterprise, <http://www.xyenterprise.com>

### *How to use this book*

*The XML Handbook* has eighteen parts, consisting of 69 chapters, that we intend for you to read in order.

Well, if authors didn't have dreams they wouldn't be authors.

In reality, we know that our readers have diverse professional and technical backgrounds and won't all take the same route through a book this large and wide-ranging. Here are some hints for planning your trip.

To start, you can get the best feel for the subject matter by reading the Table of Contents and the introductions to each part. The introductions are less than a page long and usually epitomize the subject area of the part in addition to introducing the chapters within it.

Part One contains introductory tutorials and establishes the terminology used in the remainder of the book. Please read it first.

Parts Two through Fourteen cover different application domains. The chapters are application discussions, case studies, and tool category discussions, plus some introductory discussions and tutorials. You can read them with only the preceding parts (especially Part One) as background, although technical readers may want to complete the remaining tutorials first.<sup>2</sup>

Those can be found in Parts Fifteen through Seventeen. We strove to keep them friendly and understandable for readers without a background in subjects not covered in this book. Tutorials whose subject matter thwarted that goal are labeled as being a tad tougher so you will know what to expect, but not to discourage you from reading them.

- 
2. The chapters in these parts are illustrated by their sponsors' experiences and products. The organization into parts only classifies the subject matter of the chapters; there is no attempt to classify the products. A tool used to illustrate the content management part, for example, might be an appropriate choice for a portal or a publishing system.

Part Eighteen contains resources: guides to the CD-ROM and to public XML vocabularies, an acronym dictionary, and a guide to the other books in this series.

### *Acknowledgments*

The principal acknowledgment in a book of this nature has to be to the people who created the subject matter. In this case, I take special pleasure in the fact that all of them are friends and colleagues of long standing in the SGML community.

Tim Bray and C. Michael Sperberg-McQueen were the original editors of the XML specification, later joined by Jean Paoli. Dan Connolly put the project on the W3C “to-do list”, got it started, and shepherded it through the approval process.

But all of them agree that, if a single person is to be thanked for XML, it is Jon Bosak. Jon not only sparked the original ideas and recruited the team, but organized and chaired the W3C XML Working Group.

As Tim put it: “Without Jon, XML wouldn't have happened. He was the prime mover.”

But before there could be an XML, or even an SGML, there had to have been ML. I continue to be grateful to Ed Mosher and Ray Lorie, my co-inventors – and namesakes – of the first markup language, IBM's GML.

Acknowledgment is due also to Norm Scharpf and the Graphic Communications Association for their support and community-building efforts on behalf of XML, a continuation of their “from day one” support of SGML.

Regarding the content of the book, Paul and I would like to thank Jean Paoli, Jon Bosak, G. Ken Holman, Bob DuCharme, Eliot Kimber, Andrew Goldfarb, Lars Marius Garshol, Lilia Prescod, and Steve Newcomb for contributing great material; Bryan Bell, inventor of MIDI, education video producer, and information system architect, for his advice and support; Steve Pepper and Bob DuCharme for talent-spotting; and Priscilla Walmsley for her insights into XML Schema.

We also thank Lilia Prescod, Thea Prescod, and Linda Goldfarb for serving as our usability test laboratory. That means they read lots of chapters and complained until we made them clear enough.

Prentice Hall PTR uses Adobe FrameMaker and other Adobe graphic arts and publishing software to produce most of its books. We thank Jennifer Brieger of Adobe for providing Paul and me with copies. We also thank David Turner of Microsoft for providing copies of its drawing tools.

Paul and I designed, and Paul implemented, an SGML-based production system for the book. It uses James Clark's Jade DSSSL processor, FrameMaker+SGML, and some ingenious FrameMaker plug-ins designed and implemented by Doug Yagaloff of Caxton, Inc. We thank Doug, and also Randy Kelley, for their wizard-level FrameMaker consulting advice.

But a great production system is nothing without a great production team. Faye Gemmellaro supervised the project for the publisher; we thank her for her dedication, attention to detail, and unfailing kindness and patience. We also appreciated her excellent rapport with the cover artist, Anthony Gemmellaro, who responded to our constantly changing complex requirements with inspired designs.<sup>3</sup>

We also thank Peter Snell for exercising his formidable logistic and diplomatic skills to develop our two CD-ROMs from the contributions of several hundred people.

While I do appreciate the value of computerized document processing,<sup>4</sup> I also appreciate the skill that only human domain authorities can bring to index construction. For this book, the authorities are Audrey and Ron Turner and their team at Soph-Ware Associates, distinguished XML consultants, developers, and trainers with whom I've worked on many projects over the years.<sup>5</sup>

For past editions of this book Andrew Goldfarb served as art director and artist-on-demand. For this edition he assumed the role of Managing Editor, adding the sponsorship program and other editing and production tasks to his responsibilities.<sup>6</sup>

Mark Taub, our Editor-in-Chief at Prentice Hall PTR, first proposed that I edit a book series on markup languages in 1993, long before markup became cool. I thank him again for his continued help, encouragement, and guidance, for both the series and *The XML Handbook*.

- 
3. And thanks again to Camille Trentacoste, our previous production supervisor. Although she now has other responsibilities for Prentice Hall, she still took time to give us another fishing lesson!
  4. Does that surprise you?
  5. In fact, they co-wrote the first book in my Prentice Hall series, *ReadMe.1st: SGML for Authors and Editors*, among many other books and articles.
  6. Andrew was once a best-selling author in his own right. For a brief period his *The Ballad of a Slow Poisoner* outranked both Stephen King and The Bible on the Barnes and Noble e-book best-seller list. Check out his non-XML art at: <http://www.slowpoisoners.com/flyers>



I'd also like to acknowledge a major debt to four people who supported and encouraged my efforts to develop and popularize markup languages. They epitomize vision and leadership in technical management. Norm Pass and Bobby Lie were my managers at IBM during most of the 25 years that I worked on GML and SGML. Yasufumi Toyoshima and Charles Brauer, then of Fujitsu Network Communications, were my consulting clients for six years following my retirement from IBM in 1994. They saw the potential for a Web-friendly, grammatically simple SGML subset long before I did – or anyone else I know.

As the senior author, I gave myself this preface to write. I'm senior because Paul's folks were conceiving him about the same time that I was conceiving SGML. In return, Paul got to write the history chapter, because for him it really is history.

This gives me the opportunity to thank Paul publicly for the tremendous reservoir of talent, energy, and good humor that he brought to the project. The book benefited not just from his XML knowledge and fine writing skills, but from his expertise in SGML, Jade, and FrameMaker that enabled us to automate the production of the book (with the previously acknowledged help from our friends).

Thanks, Paul.

*Charles F. Goldfarb*  
*Saratoga, CA*  
*November 14, 2001*

# Foreword

When HTML came onto the scene it sparked a publishing phenomenon. Ordinary people everywhere began to publish documents on the Web. Presentation on the Web became a topic of conversation not just within the computer industry, but within coffeehouses. Overnight, it seemed as though everyone had a Web page.

I see the same phenomenon happening today with XML. Where data was once a mysterious binary blob, it has now become something ordinary people can read and author because it's text. With XML, ordinary people have the ability to craft their own data, the ability to shape and control data. The significance of this shift is difficult to overstate, for not only does it mean that more people can access data, but that there will undoubtedly be more data to access. We are on the verge of a data explosion. One ignited by XML.

By infusing the Web with data, XML makes the Web a better place for people to interact, to do business. XML allows us to do more precise searches, deliver software components, describe such things as collections of Web pages and electronic commerce transactions, and much more. XML is changing not only the way we think about data, but the way we think about the Web.

And by doing so, it's changing the way we think about the traditional desktop application. I have already witnessed the impact of XML on all

types of applications from word processors and spreadsheets to database managers and email. More and more, such applications are reaching out to the Web, tapping into the power of the Web, and it is XML that is enabling them to do so. Gone are the days of the isolated, incompatible application. Here are the days of universal access and shared data.

I joined Microsoft in the summer of 1996 with great faith in the Standard Generalized Markup Language (SGML) and a dream that its potential might one day be realized. As soon as I arrived at Microsoft, Jon Bosak of Sun Microsystems and I began discussing the possibility of creating an XML standard. Jon shared my enthusiasm for a markup language such as XML, understanding what it could mean to Web communication.

My goal in designing an XML standard was to produce a very simple markup language with as few abstractions as possible. Microsoft's success is due in no small part to its ability to develop products with mass-market appeal. It is this mass-market appeal that I wanted to bring to XML. Together with Jon and other long-time friends from the SGML world, C.M. Sperberg-McQueen, James Clark, Tim Bray, Steve DeRose, Eve Maler, Eliot Kimber, Dave Hollander, Makoto Murata, and Peter Sharpe, I co-designed the XML specification at the World Wide Web Consortium (W3C). This specification, I believe, reflects my original goals.

It was truly an exciting time. For years, we had all been part of a maverick band of text markup enthusiasts, singing its praises every chance we had, and before us was an opportunity to bring XML into the mainstream, maybe even into the operating system. At last, we were getting our chance to tell the World of the thing we had been so crazy about for all this time.

By the fall of 1996, many groups inside Microsoft, including Office, the Site Server Electronic Commerce Edition, the Data Access Group, to cite a few, were searching for an open format to enable interoperability on the Web. It was then that I began working with the managers of Internet Explorer 4, with the passionate Adam Bosworth, with Andrew Layman, with Thomas Reardon, to define the Channel Definition Format (CDF). CDF, the first major application of XML on the Web, became an immediate and incredible success, and XML started catching on like wildfire across the Web.

I remember those weeks and months that followed as a time where it seemed that every day another new group within Microsoft began coding applications using XML. Developers, left and right, were turning on to XML. They frenetically began to develop applications using XML, because XML gave them what they wanted: an easy-to-parse syntax for representing