by Brad Eden

Although this book is stated to be geared specifically towards librarians, the author has provided an overview of the eXtensible Markup Language (XML) for non-librarians as well. The text provides basic information as well as important applications and uses for XML in today's current information environment. The relationship of XML to SGML and HTML is discussed, as well as the various components of the markup language. The structure of an XML document is explained, along with different types of style sheets that offer various options for formatting and presenting XML documents for reading and processing. The linking and pointing components of XML are illustrated as well. The importance of the processing and import of XML data, especially as related to electronic commerce, will accelerate XML's adoption and widespread acceptance, according to the author. The book comes with a glossary and bibliography, and states that contents will be updated at http://newtechnologypress.com/updates/xml.html (http://newtechnologypress.com/updates/xml.html).
XML is a meta language, similar to metadata, except that it provides the syntax that allows users to create their own markup language and vocabularies for their own needs. It was approved by the World Wide Web Consortium (W3C) on February 10, 1998. The goals were to make it compatible with SGML, capable of supporting a wide variety of applications and easy-to-write programs that would process documents in XML, easily transferred over a network, and legible to humans yet machine-readable. If readers wish to keep up with the current status of XML in W3C, they should visit http://www.w3.org/TR/.

Chapter 1 is a basic introduction to XML and its relationship to SGML. The author discusses Data Type Definitions (DTDs), DTD alternatives, and how the MARC format can be used as a DTD in XML. The differences between "valid" and "well-formed" XML, as well as current applications that incorporate XML, are provided.

Chapter 2 discusses the variety of tags that are used in XML documents. Both the logical structure and the physical structure of an XML document are examined in some detail. The XML declaration, DTD, elements, and attributes are all explained under the logical structure of an XML document. Character, general, and parameter entities are discussed under the physical structure of an XML document, as are notation declarations.

Chapter 3 examines how XML documents are structured, and how important it is that markup and style rules are separated. A brief explanation of the Document Style Semantics and Specification Language (DSSSL) is given, and Cascading Style Sheets (CSS) are explained. The importance of flow objects is discussed in relation to the construction of XML style sheets. The author provides some examples of current tools available in the marketplace to create XML style sheets, and what the W3C is doing to accommodate multimedia elements in XML.

XML's capabilities for pointing and linking are the main topic of Chapter 4. SGML does not support hyperlinking, which is why HTML has become so popular with Web designers. XML's Linking Language (XLL) provides this function and is divided into three specifications: XLink, XPointer, and XPath. The author provides some background and comparison on the similarities and differences between HTML links and XML links. Simple and extended links in XML, as well as their various attributes and meanings, are detailed. The three ways to address a link with XPointer are examined (absolute keywords, relative keywords, and string matching). Finally, the different node types, absolute and relative location paths, and navigation recognized in Xpath, are discussed.

XML processing software, or parsers, is the topic of Chapter 5. As a result of XML's ability to manipulate data, an XML processor can follow two different approaches. The first is event-driven processing, which uses the Simple Application Programming Interface (API) for XML (SAX). It works fast and doesn't consume a lot of memory, but its disadvantage is that it is impossible to look ahead to make decisions based on information that comes later in the data stream. The second is tree-manipulation processing, which uses the Document Object Model (DOM). This approach requires two passes through the data, and takes care of the disadvantage of event-driven processing, but it is more difficult to build a tree and then navigate it. It requires more memory and is slower in processing time. The author addresses both SAX and DOM, and ends the chapter with a discussion of current applications incorporating these two APIs.

Document management issues related to the implementation of XML are examined in Chapter 6. Different database architectures and their utility for storing XML documents are discussed, along with managing resources on a network and making networked information more interoperable.
Chapter 7 focuses on the application of XML and its potential for electronic commerce. The author reviews the Electronic Data Interchange (EDI) standard, the importance of metatags and DTDs for improving functionality of business transactions and interpreting data structures, and then looks at how XML can assist in these endeavors. A number of current electronic commerce consortia and collaborations are described at the end of the chapter.

Chapter 8 is basically a "getting started" manual for anyone who is interested in XML coding. Four classes of current software tools are examined, and short abstracts on various products in each category are listed. These include content development programs, application development packages, databases, and schema development kits.

For those librarians who already know and use HTML, this book is an essential manual for the coming transition to XML. If one already understands the structure of HTML, then the technical language contained in this book should be understandable. For those who have never worked in HTML, or in any other markup language, some of the in-depth discussion and description of the workings of XML may prove to be a little hard to understand. I found the last few chapters to be most useful to those who wish to find information on the various software packages available to assist in XML coding, and would like some discussion of their benefits and disadvantages. Overall, this book is a welcome addition and indeed a necessity for anyone involved in the description and organization of information, given that XML is forecasted as being the "next generation HTML" for the Internet. While HTML coding is only concerned with display, XML coding provides language for both display and content, and is already proving to be important in the construction and acceptance of many metadata standards.

Brad Eden, Ph.D., Head, Bibliographic and Metadata Services, University of Nevada, Las Vegas, beden@ccmail.nevada.edu.

Copyright (c) 2002 by Brad Eden. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at beden@ccmail.nevada.edu (mailto:beden@ccmail.nevada.edu).


by Shirl Kennedy

I currently have in my bathroom medicine cabinet a container of Right Guard Xtreme Sport deodorant (http://www.gillette.com/products/grooming_toiletries.asp). In one of my kitchen cupboards, I have a box of Ritz Bits Sandwiches - Xtreme Cheese (http://www.nabiscoworld.com/RitzBits/Default.htm). My younger son, searching the Internet for NFL 2K2 video game hints, came across a site called Xtreme Cheats ("Game Cheats to the Xtreme!" - http://www.xtreme-cheats.com/). And in my favorite bar, it's common to see one of the TVs tuned to Extreme Sports events on ESPN (http://expn.go.com/), featuring dangerous athletic endeavors designed to provoke cringing in someone my age.
Kind of makes you wonder about something called Extreme Programming (XP), doesn't it?

At the Web site "Extreme Programming: A Gentle Introduction," (http://www.extremeprogramming.org/), we find out that this has nothing to do with marketing hype or skateboards. It is, in fact, a software methodology -- a fancy term for a "set of rules and practices used to create computer programs." Do not attempt to read or even browse through Extreme Programming in Practice until you first acquire a good understanding of what Extreme Programming is -- an approach to software development that emphasizes teamwork, simple practices and ongoing communication, and that focuses on customer satisfaction. The emphasis on teamwork and communications allows for a quick shift in direction, if customer requirements change or are not being met in some way.

This book is one of seven titles in Addison-Wesley's XP Series, (http://cseng.aw.com/catalog/series/0,3841,13,00.html), and it is definitely not the first one you should pick up. Rather than introducing or explaining the topic, it is more of a protracted case study, packed with lines and lines of code. Essentially, the authors have put together a journal of their efforts in using XP practices to program a user registration system for their training/consulting firm's Web site -- a three-week project. Frankly, if you're not a programmer, you're probably not going to get a whole lot out of this book.

The first couple of chapters describe the specific problems the programmers plan to solve, and sketch out the direction of the book. The third chapter, an introduction to Extreme Programming, is readable by just about anyone and is a pretty good discussion of the philosophy and structure of an XP project. It stresses the involvement of the customer/client in all phases of the project, and the fact that "all production software is written by pairs of programmers," rather than by one or more geeks keyboarding in splendid isolation in their individual cubicles.

The fourth chapter is an account of sitting down with the "customer" -- in this case, their firm's vice president -- and parsing "user stories" to determine how the Web site was currently functioning and which mechanisms and processes needed to be changed. The fifth and sixth chapters explore the project planning phase and, finally, delineate the specific steps in the plan.

In the seventh chapter, the heavy coding examples begin, and these continue pretty much up to Chapter 13, where the "first iteration" of the new registration system is functional. The customer then tests it out and reports on what he does and doesn't like about it -- ultimately translated into more "user stories." The team discusses what went wrong and why, and determines how to go about fixing the problems.

In Chapter 14, the team works through to the final release, which takes another two "iterations," and gets the registration system up and running on the live site. The authors briefly speculate on how the project would have gone differently if they had not been using the XP methodology -- the main difference being that it would have taken a lot longer. In the concluding Chapter 15, the authors come up with a list of "lessons learned," and discuss how these might be carried over to larger XP projects. The appendix contains all the source code (Java) for the first iteration of the project. In theory, if you sit down and type it out line by line, you will end up with a registration system for your Web site. Not being a programmer myself, however, I am not qualified to evaluate the code on its merits.

This is not a book for someone who has never used XP methodology (or managed a team using it), and is looking for guidance on how to get started. My gut feeling is that it would be most valuable to someone already involved in or familiar with XP, who might be interested in how somebody else is using it. But those
without a programming/software development background will likely be left dazed and confused by this book.

References:

(http://www.extremeprogramming.org/)

(http://www.xprogramming.com/)


by Michael B. Spring

It is a little hard to write a comprehensive review of a reference card, but it ends up that the challenge has been an enjoyable one. In some ways, this review could exceed in length the 8.5-inch by 45-inch reference card made up of 13 3.5 x 8.5-inch folded panels!

The Bash or "Bourne Again SHell" is a thing of beauty, combining the best features of many of the other shells and making them exceptionally easy to use. While there are still a few reasons to use the Korn or C shells on occasion, Bash tends to provide most of the specialized features of these shells. But the richness of Bash makes its full power difficult for novices -- and even some experts. Thus, one is always looking for an authoritative and understandable reference. The first question one asks about the card is, how easy is it to find information that is needed using the reference. While the card generally follows the order with which topics are addressed in the man pages (manual pages) on Bash, a masterful job has been done making what is covered on each panel visible at a glance. In addition, the contents on the front cover make it easy to identify the location of needed information. As someone who has a lot of trouble getting the right part of a folded map in front of me, this reference card was a real delight. To find panel 8 or 19 was a snap.

Browsing is also a delight. The card can actually be read from front to back and it makes sense. This reader was amazed to find that in several cases the condensation was actually clearer than the parallel more verbose treatment in the man pages. Indeed, a couple features of Bash that had skipped my attention over
five years of use popped out in scanning the card (if you must know, brace expansion, variable substitution, and directory stack operations). The synopsis of the built-in commands and their options is clear and to the point.

If there is a weakness on the card, it may be the lack of cross-references. It would appear to this reviewer that an abbreviated panel/topic cross-reference (each panel has one or more topic boxes) could have been included where significant dependencies existed. This would allow the reader to get to a see-also topic without having to go back through the table of contents. While it won't help the novice to learn Unix, intermediate to experienced Unix users who are using Bash early on -- Bash is the default shell for Linux -- will find the card as useful as a larger reference book, and at $4.50, it provides what is needed and nothing more clearly and concisely for a good price.

Michael B. Spring is an Associate Professor of Information Science and Telecommunications at the University of Pittsburgh. He can be reached at spring@imap.pitt.edu.

Copyright (c) 2002 by Michael B. Spring. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at spring@imap.pitt.edu.


by Kwan-Yau Lam

The purpose of this book (hereafter referred to as "the Toolkit") is quite clear from its title. It is intended for librarians and professionals who are charged with the responsibility of providing patrons with Internet training or instruction. The book contains 12 training modules that are based on the author's work at a public library developing Internet training workshops.

The organization of the Toolkit is clear and straightforward. The first four modules cover four very fundamental areas for beginning Net-surfers: personal computer basics, Internet and Web navigation, Web searching, and Web site evaluation. The remaining eight modules provide an introduction to various Internet topics: e-mail, chatting, Web site development and maintenance, multimedia, digital cameras, online shopping, online auctions, Web resources, and online safety tips for young children.

The Toolkit comes with a CD-ROM, which is perhaps the main strength of the book. The CD-ROM contains fliers, handouts, scripts, and PowerPoint slides for all the modules in the book. Librarians and trainers can easily copy those files from the CD-ROM and make appropriate changes to suit specific instructional needs. It therefore provides a lot of convenience, and it is in fact what transforms the book into an actual toolkit for Internet training and instruction.

The Toolkit CD-ROM provides yet another advantage. The multimedia materials and PowerPoint slides stored on it allow librarians and trainers to present information in ways impossible in print format. Needless to say, to many people, it is much more boring reading all the black-and-white, motionless slides and scripts in the book than actually viewing a multimedia PowerPoint slide show with sound and animated effects.
Despite the above-mentioned strengths, the Toolkit does have its own limitations. First, it should be noted that there are actually two groups of target audiences -- the target audience of the book, and that of the modules. As noted earlier, the Toolkit is intended for librarians and training professionals who offer Internet workshops. These librarians and trainers are the target audience of the book. On the other hand, the target audience of the modules is comprised of people who attend the workshops. Since the modules are built upon the author's work as a training specialist at a county public library, they seem to be very much oriented toward public library patrons. As a result, not all librarians may find all the 12 modules useful for their work. For example, many academic librarians may find Module 6 (Shopping the World Wide Web -- The Internet Consumer Guide) and Module 11 (Selling and Saving: Exploring WWW Auctions) to be of very little use, because they do not have any practical need to offer online shopping or auction workshops for college or graduate students. As an introductory Internet guide, the Toolkit may perhaps better serve its purpose by including more general topics that appeal to a broader audience rather than personal interests like Web auctions.

It is not always easy to strike a good balance between breadth and depth, especially when developing introductory training materials. The Toolkit suffers from this drawback to a certain extent as well. To be sure, all the modules do contain a lot of useful information. Yet, there are still areas that do not seem to be covered adequately. For example, newsgroups and discussion lists are only mentioned briefly in Module 2 (Navigating the Internet and the World Wide Web) as parts of the Internet. Even though they are helpful information resources for a lot of people, they are not discussed in great detail and unlike e-mail and chatting, they do not have their own separate modules in the book. Another example would be after going through Module 1 (Introducing the Personal Computer), a computer novice may still not know how to actually make a modem connection to the Internet from the Windows Desktop.

The modem connection example above brings up another point. This reviewer has once seen a computer simulation of dialing up the Internet, complete with sound effects. Such a simulation can give the learner a near-hands-on experience, which is important because it is not always possible to provide learners with actual hands-on learning opportunities. As a matter of fact, it is not always possible for workshop attendees to actually dial up the Internet since many libraries already have direct connections. Without any doubt, computer simulation is a valuable training and learning tool. Adding computer simulation programs to the Toolkit CD-ROM would definitely add more educational value to the book and would enhance the training and learning process.

Finally, it should be noted that there are a lot of computer terms and jargon found in every module. Although the author has done a wonderful job explaining this jargon in very general, easy-to-understand terms, there are occasions in which further elaboration and clarification seem necessary. As an example, the author describes the CPU (central processing unit) of a personal computer as the computer's main box which holds the "hard drive, RAM, processor, and inputs and outputs" (p.5, Slide 5, Module 1). While this description is quite common and widespread, it is not really accurate (NetLingo Dictionary, n.d.) More technically and more correctly speaking, a personal computer's CPU is the single chip, often known as the microprocessor, that is the computing part of the computer and that communicates with peripheral devices (Parker & Morley, 2002; TechEncyclopedia, n.d.) This distinction may sound trivial to many people. However, for many librarians and particularly those in academic or research libraries, it is very important to provide accurate information.

In short, this Toolkit can be a useful resource for librarians and professionals who need to develop and provide basic Internet training or instruction. Contents of the modules are pretty straightforward and descriptive. All of the modules focus only on the basics, even though pointers to further information and
readings are sometimes provided. Because of the basic nature of this Toolkit, librarians and trainers preparing for Internet workshops beyond the beginner level would probably want to find another book with which to start.

References:


Kwan-Yau Lam (klam@ccc.edu (mailto:klam@ccc.edu)), Assistant Professor / Librarian, Truman College, City Colleges of Chicago.

Copyright (c) 2002 by Kwan-Yau Lam. This document may be reproduced in whole or in part for noncommercial, educational, or scientific purposes, provided that the preceding copyright statement and source are clearly acknowledged. All other rights are reserved. For permission to reproduce or adapt this document or any part of it for commercial distribution, address requests to the author at klam@ccc.edu (mailto:klam@ccc.edu).

About TER