

TER Volume 4, Issue 11, December 1, 1997



Volume 4, Issue 11, December 1, 1997

Telecommunications Electronic Reviews (TER) is a publication of the Library and Information Technology Association.

Telecommunications Electronic Reviews (ISSN: 1075-9972) is a periodical copyright © 1997 by the American Library Association. Documents in this issue, subject to copyright by the American Library Association or by the authors of the documents, may be reproduced for noncommercial, educational, or scientific purposes granted by Sections 107 and 108 of the Copyright Revision Act of 1976, provided that the copyright statement and source for that material are clearly acknowledged and that the material is reproduced without alteration. None of these documents may be reproduced or adapted for commercial distribution without the prior written permission of the designated copyright holder for the specific documents.

Contents:

- REVIEW OF: S. Keshav. An Engineering Approach to Computer Networking: ATM Networks, the Internet and the Telephone Network. by Alicia Abramson
- REVIEW OF: Ron Ben-Natan. Objects on the Web: Designing, Building, and Deploying Object-Oriented Applications for the Web. by Craig S. Booher
- REVIEW OF: Pratik Patel and Karl Moss. Java Database Programming with JDBC. by Genevieve Engel
- About TER

 [ter issues \(/lita/publications/archive/ter\)](/lita/publications/archive/ter)

REVIEW OF: S. Keshav. An Engineering Approach to Computer Networking: ATM Networks, the Internet and the Telephone Network. Reading, MA: Addison Wesley, 1997.

by Alicia Abramson

Having recently taken a graduate-level introductory course in telecommunications, I vividly recall my classmates' responses to our textbook: "too dry," "too detailed," "too technical for students in the Information Systems program" (as opposed to the Computer Science program), and the frustrated cry, "how are we supposed to remember all of this stuff for our exam?" I had to agree with this sentiment based on my own motivation for taking the course: to get an overview of modern telecommunications systems and how they work and interact with each other, and to develop a methodology for examining new and emerging telecommunications technologies and evaluating them for possible use in a real life setting. The textbook, which was the focal point for the course, delved a little too deeply for my own tastes into the details and inner workings of such things as Protocol Data Units (PDUs), Asynchronous Transfer Mode (ATM) cells, and Transmission Control Protocol/Internet Protocol (TCP/IP).

Despite these complaints, the textbook in the course, *Data and Computer Communications* by William Stallings, is considered one of the standard texts in the field and, by virtue of the depth of technical detail and scope of coverage, a difficult book to measure up to in stature. [1] In my opinion, however, it is a lot to digest in a one semester introductory course. I would argue that S. Keshav's *An Engineering Approach to Computer Networks: ATM Networks, the Internet and the Telephone Network* is a suitable alternative for such a course. With the smaller scope of Keshav's book, students will emerge feeling they know more about a few major computer network protocols and technologies as opposed to gaining a little knowledge about many.

Keshav does not cover all of the topics or go into the same level of depth about all of the networking protocols, models, and operational details that Stallings covers (for example Local Area Network, or LAN, coverage is slim, and Frame Relay and ISDN (Integrated Services Digital Network) are not covered at all). Readers of this text, however, will come away from the book with a good understanding of how the three focal networks (ATM, Internet, and telephone) work.

More importantly, the book provides a solid grounding in the real-life design and implementation issues involved in building computer networks. Keshav provides students with a clearly written text, numerous real-world examples, and a balance of background information, technical information, and framework for developing a problem-solving approach to the design and implementation of computer networks. While Keshav's audience is first-year graduate level computer science and engineering students, as already stated, those coming from a more general, rather than technical, background will also benefit from this text.

The book has 660 pages and is divided into 16 chapters that are grouped into three sections: an introductory section which focuses on the background and history of telephone and computer networks (chapters one through four); a middle section that focuses on protocols and network functionality such as switching, scheduling, addressing, routing, and error and flow control (chapters five through fourteen); and the final section which examines the functionality covered in the previous chapters in relation to actual ATM, Internet, and telephone protocol implementations (chapters fifteen and sixteen).

Each chapter ends with a set of review questions and exercises. The author also provides a Web site (<http://www.cs.cornell.edu/skeshav/real/index.html>) that has additional and more advanced exercises including a downloadable "multithreaded, packet-level network simulator that allows users to simulate arbitrary protocols written in C." (p. xviii) A quick review of the Web site shows that the simulator runs on most major UNIX platforms. Additional material on the Web site includes errata and access for instructors to the answers for the book's exercises and review questions.

Keshav's intention from the outset is to make this a practical book. He defines this practical, engineering approach to computer networks in the preface: it involves "identifying the fundamental constraints on the problem, mak[ing] reasonable real-world assumptions, and then examin[ing] several alternative solutions, trading off their pros and cons." (p. xiii) True to this practical approach, throughout the text Keshav pushes his readers to think not only in terms of technical aspects of computer networking and the best technical solution to networking problems, but also in terms of economics and what technical trade-offs are acceptable in relation to the cost of a particular solution. It is Keshav's consistent provision of background information and real-life examples--and thus context--throughout the book that sets his book apart from other textbooks, such as Stallings', that rely almost exclusively on abstractions to teach complex concepts.

"Section 1: Introduction" is composed of an introductory chapter where the entire book is outlined and three chapters covering the "Concepts, History and Challenges" of the telephone, Internet, and ATM networks respectively. In these chapters, Keshav provides readers with historical background outlining the development of the three networks, their underlying structures and functionality, and their limitations and areas of potential growth.

In addition, Keshav introduces concepts and terminology that will be used throughout the remainder of the book, as well as comparing and contrasting the different networking styles embodied in the three networks discussed. The contrast between the telephone network and the Internet supplies the rationale for focusing on ATM as the third main network for examination in the book.

For example, the telephone network is connection-oriented and places all of the intelligence in the network itself, while the Internet is a packet-switching network that relies on the intelligence of the end-stations. While these designs met the original demands placed on them, they are less adequate for current demands, such as the delivery of high-bandwidth, time-sensitive, multimedia content to the desktop and the home. Keshav chooses ATM as an example of computer network design that attempts to combine the best features of these earlier successful network designs, navigating around their weaknesses to produce a more technologically and economically efficient network design capable of handling these current demands. Features such as different levels of Quality of Service (QOS) and guaranteed delivery of multimedia are two notable improvements.

"Section 2: Tools and Techniques" comprises the bulk of the book. Here, Keshav is methodical in developing a framework for examining network design and teaching his readers about the challenges of actually implementing networking protocols in a dynamic environment. The first chapter in this section looks at protocol layering and the rationale for it, as well as the advantages and limitations of layered models. While most of the chapter covers the layers in the seven-layer Open Standards Interconnect (OSI) model, Keshav, breaking from tradition with other textbooks in the field, lays out the facts that most people familiar with the layered model debate recognize: the IP protocol stack won the battle and is the de facto standard, and the OSI model lost. Oddly, while he mentions this fact, he leaves most discussion of IP to chapter 15, thus missing the chance to contrast its differences with OSI.

Keshav moves on to a chapter discussing system design which he considers to be both an art and science. While textbooks like Stallings' tend to focus on how systems work without much attention to how or why they are developed, Keshav is more interested in getting students to think about what's involved in putting systems together, with a focus on five main constraints on design: time, space, computation, money, and labor. Throughout the chapter, Keshav illustrates different techniques available for working around these constraints to achieve optimal system performance.

In the remaining eight chapters of section two, Keshav devotes a chapter each to the main technical problems faced in the operation of a computer network: multiple access, switching, scheduling, naming and addressing, routing, error control, flow control, and traffic management. By virtue of its recent publication, Keshav's book is able to cover some topics that older computer networking texts don't, such as multi-cast routing over both the Internet and LANs to allow groups to receive multimedia broadcasts, and Resource Reservation Protocol (RSVP), which has the potential to bring levels of service and guaranteed reservation of bandwidth to the Internet.

Keshav saves the last two chapters in "Section 3: Practice" for a more in-depth discussion of specific protocol implementations in the three focal networks. The protocols explored include Synchronous Optical Network (SONET) over the telephone network; IP, Internet Control Message Protocol (ICMP), and Hypertext Transfer Protocol (HTTP) over the Internet; and IP-over-ATM, Service Specific Connection Oriented Protocol (SSCOP), and ATM Adaptation Layer (AAL) over ATM networks.

On a critical note, perhaps the biggest drawback to Keshav's decision to focus on ATM as his third example type of network is the many difficulties ATM continues to encounter as a solution to the problem of efficient, high-bandwidth delivery, such as ATM/IP interoperability issues and the high costs of integrating ATM into existing networks or switching to ATM altogether. Keshav treads a thin line between highlighting ATM as an one example of a design solution and looking to ATM as the solution to the bandwidth and guarantee-of-delivery problems.

As noted in a commentary in a recent issue of Telecommunications Electronic Review, "The intellectual challenge is to avoid jumping to the conclusion that because ATM can offer this feature that ATM will never run out of bandwidth...." [2] And that it is the only solution. Keshav might have included at least brief discussions of Asynchronous Digital Subscriber Line (ADSL) and Frame Relay, talking about how they figure into the evolution of the telephone network and Internet into a truly high-speed network.

One final criticism of this book is the sometimes sloppy layout. Examples in the text are separated on the page by a leading and trailing horizontal line. This gets very distracting in chapters with many examples, such as in chapter six where it becomes difficult to follow the regular text when it is interspersed between more than one example on a page. This and some other minor inconsistencies in page layout detract from an otherwise clearly written text.

Notes

[1] Stallings, William S. (1997). Data and Computer Communications. Fifth edition. Upper Saddle River, NJ: Prentice-Hall.

[2] Wilson, Thomas C. (1997, July). Life as Bandwidth|Networking is Life. *Telecommunications Electronic Review*, 4(7). <http://www.lita.org/ala/lita/litapublications/ter/terv4n7july.htm#wilson> (<http://www.lita.org/ala/lita/litapublications/ter/terv4n7july.htm#wilson#wilson>).

Alicia Abramson (aabrams@american.edu (mailto:aabrams@american.edu)) is Head of Library Systems at American University in Washington, DC.

Copyright © 1997 by Alicia Abramson. 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 aabrams@american.edu (mailto:aabrams@american.edu).



REVIEW OF: Ron Ben-Natan. Objects on the Web: Designing, Building, and Deploying Object-Oriented Applications for the Web. New York: McGraw-Hill, 1997.

by Craig S. Booher

In the past three years the World Wide Web and, more generally, the Internet have been increasingly touted as the information superhighway. While the number of information resources available over the Internet continues to explode, a new frenzy of activity is beginning to surface on the Web--accessibility to fully functional applications. Under this new paradigm, the Internet becomes the vehicle through which fully functional applications are deployed. Web browsers provide the client environment in which platform-independent applications can be dynamically downloaded upon request.

Objects on the Web focuses on the technological foundations for Web applications. As its title implies, this text considers much, if not all, of the Web technology from an object-oriented point of view. In fact, one of the early chapters provides a general overview of object orientation. Here the reader is introduced to the basic concepts, history, and methodologies (not to mention acronyms) of object technology. This chapter is essential (although perhaps not sufficient) for anyone not previously exposed to object-oriented principles.

Generally speaking, the book can be divided into three discussion areas. One section primarily focuses on such basic Web components as HTML (Hypertext Markup Language) and HTTP (Hypertext Transfer Protocol). A second section with six chapters discusses application development tools, including Java, JavaScript, VisualWave, and WebObjects. Finally, three appendices (representing over thirty percent of the book) provide vendor documentation for VisualWave, NetDynamics, and OrbixWeb.

The section covering basic Web components consists of seven brief (usually about 20 pages) chapters, each dedicated to one of the following topics: history and organization of the Internet; the aforementioned introduction to object orientation; HTTP; HTML; forms, CGI (Common Gateway Interface), and server extensions; VRML (Virtual Reality Modeling Language); and security. Obviously, each chapter could be, and has been, the subject of entire volumes.

Why, then, are they included? With the exception of the chapter introducing object-oriented technologies, none of the chapters is critical to the understanding of the object-oriented discussions that follow. In fact, one would assume that the intended audience--systems managers, systems designers, and application developers--would already possess reasonable fundamental knowledge in each of these basic Web technologies.

Regardless of the reason for their inclusion, these chapters do provide accurate, well-organized, broad coverage (mostly at an altitude of 10,000 feet with regular dips down to ground level to provide illustrative examples) of these areas. Ben-Natan's liberal use of code examples and sample screen dumps helps clarify many of the discussions. His coverage of the various standards, their development history, and the corporate players helps the reader better understand the environment in which the existing technologies were birthed or sanctioned.

The second major section of the book presents a discussion which is more tightly aligned with the author's stated objectives. Here the reader begins to learn about the tools used to deliver applications via the Web.

Three chapters are devoted to Java and JavaScript. For each language, Ben-Natan begins by providing a development history of the language, pointing out Sun's significant contributions in this area. Subsequently, he outlines the basics for each language, its object-oriented features, and its syntax. Finally, he demonstrates the use of Java applets to deliver applications through Web browsers. As with the basic technologies chapters, the author's style is to cover the distinguishing features, highlight critical components and fundamental concepts, and use code examples to illustrate text discussions.

Ben-Natan continues his exploration of the Web as an application delivery platform by discussing three commercially available products: VisualWave from ParcPlace/Digitalk, WebObjects from NeXT, and Spider Technology's NetDynamics. All these products use a three-tier approach for building an application framework. They view the Web as a front-end/back-end architecture, having a Web server sitting between the browser front-end and the application server back-end.

VisualWave receives the bulk of the author's attention, while the other two are each allotted two pages of text. The emphasis on VisualWave continues in a subsequent chapter where it is used to illustrate a detailed example of an application providing a simple financial analysis tool.

As with most software products, words can only go so far in describing their features and capabilities. The best understanding of the product comes from "driving" it. Fortunately, an accompanying CD-ROM contains NeXT's WebObjects Lite 2.0 for HP-UX, WINNT, and Solaris platforms. Curiously, as noted earlier, WebObjects is not one of the products covered in the vendor documentation found in the appendices.

One final chapter, entitled "Putting It All Together," briefly covers a wide variety of technologies designed to provide interfacing and bridging capabilities. By way of quick summaries (usually one to two pages), the reader is introduced to CORBA (Common Object Request Broker Architecture), JOE (Java Objects Everywhere), OrbixWeb, VisiBroker, Remote Method Invocation, JavaBeans, OLE (Object Linking and Embedding), VBScript, ActiveX, and LiveConnect.

In conclusion, let's return to the earlier question about content. One answer to that question may lie in the identity of the intended audience. Anyone with a basic exposure to Web and Internet technology will find almost a third of the book a review of fundamental knowledge. On the other hand, those looking for deep understanding of either the basic technologies or the application development technologies will find the book inadequate. By process of elimination, a middle road must be the answer.

The strength of this book lies in its breadth of coverage and summary of relevant information in encyclopedia-like fashion. Here, in one work, we have concise, crisp descriptions of most of the basic Web and Internet technologies as well as the developing Web application delivery technologies. A detailed index and over 100 references guide the reader to information within and outside the book. Unfortunately, given the rapid state of flux in this area of technology, much of the information contained in this book, especially that about the commercial products, will be behind the cutting edge within the next 12 months. So, either read this book soon, or wait for a second edition which will, hopefully, devote more attention to the application development and deployment technologies.

Craig S. Booher (cbooherr@kcc.com (mailto:cbooherr@kcc.com)) is a Technical Information Coordinator at Kimberly-Clark in Neenah, WI.

Copyright © 1997 by Craig S. Booher. 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 cbooherr@kcc.com (cbooherr@kcc.com).

[table of contents](#)



ter issues (/lita/publications/archive/ter)

REVIEW OF: Pratik Patel and Karl Moss. Java Database Programming with JDBC. Scottsdale, AZ: Coriolis Group Books, 1996.

by Genevieve Engel

Why JDBC?

The foremost question in my mind when I first heard of JDBC (Java Database Connectivity) was, "Will it enable Java front ends to library catalogs?" After studying this book, I find the answer is a qualified maybe.

If you have an Oracle, Sybase, or DB2 database and you want to provide a Web-based interface for it using Java, then the JDBC approach saves you from reinventing the wheel. Database drivers already exist for linking JDBC implementations to a variety of commercial database management systems.

If your database is a library catalog or other bibliographic database, then JDBC is either just the starting point for further tweaking (if you're committed to a Java interface) or an unnecessary layer of complexity (if you really just want some kind of Web-based access). For catalogs that don't have either SQL (Structured Query Language) or Z39.50 support, the best Java-based solution is probably the Java Telnet Applet (Web address below).

The basic purpose of JDBC is to tie Java-enabled Web pages to SQL databases. While SQL is the client/server database standard du jour in the business world, it is entirely unrelated to the Z39.50 standard heavily used for client/server database access in the library environment. If your public-access catalog supports Z39.50 protocols, you already have several existing Web-to-Z39.50 gateways from which to choose: your vendor may provide one, or you can adapt one from freely available code.

Still, if you are committed to developing a Java-based front end to your catalog, perhaps in order to gain more control over the look of the interface than is afforded by HTML (Hypertext Markup Language) alone, you may wish to consider using the existing code base of JDBC-related classes rather than developing a Java-to-Z39.50 gateway from scratch.

Why this Book for JDBC?

The book under review provides an in-depth introduction to the JDBC API (Application Programming Interface) with plenty of examples both of code and of results. Most of the book is devoted to several chapters which take the reader step-by-step through installing, using, and adapting methods and drivers. Numerous code examples are accompanied with extensive explanation of what each part of the code is doing and with remarks on how or why you might want to change the code depending on your specific environment and application. This kind of relation between the text and the code is a definite plus. The chatty and sometimes repetitive text may try the patience of the experienced Java programmer, but may serve as encouragement and valuable repetition for the novice.

Chapters and appendices in the book include an overview of JDBC; an overview of SQL; installation of JDBC drivers; constructing an applet to use JDBC features; bridging from JDBC to ODBC (Object Database Connectivity) drivers; mapping SQL data types to JDBC; formatting and displaying query results; working

with multimedia and other binary file results; database security issues; writing JDBC drivers; middleware (three-tiered) implementations; the complete JDBC API; Java fundamentals; and source code for two JDBC drivers.

Overall, the book is a valuable resource for the Java programmer implementing JDBC. A reasonably competent Java programmer could use this book as a sole source of information to develop a JDBC driver for a specific database, and could do so with minimal effort given the completeness of the sample code. Armed also with the Z39.50 specification, that programmer could possibly develop a JDBC-to-Z39.50 bridge, but some additional knowledge of SQL might be needed. The security issues covered in the book detail the limitations of JDBC in querying an arbitrary database on the Internet, so it might be difficult to develop a JDBC-based public-access interface in any event. I'll leave these problems to the reasonably competent Java programmers.

The failings of the book are on the part of the editor or publisher, not the authors. The accompanying CD-ROM, while a good idea (having the code on the CD-ROM saves the reader from having to type it in), doesn't seem to include all the files mentioned in the text. URLs (Uniform Resource Locators) given for further information are not always valid or up-to-date, and the publisher's Web site gives no updates.

The organization of the book leaves a little to be desired: an overview of SQL is presented in chapter two, while an overview of Java is presented in an appendix. The index is somewhat useful for reference after the reader is through with the book, but less useful during the initial reading: extremely specific method names are included, but basic concepts mentioned several times in the book, such as "trusted applets," are missing.

Readers wishing further information might want to consult some of these resources in addition to, or instead of, those listed in the text:

JDBC API:

<http://www.javasoft.com/products/jdbc/index.html> (<http://www.javasoft.com/products/jdbc/index.html>)

Choosing a JDBC Driver (commercial JDBC product site, but has further security notes):

<http://www.weblogic.com/whitepapers/jdbc.html> (<http://www.weblogic.com/whitepapers/jdbc.html>)

Integrating Z39.50 in the Web, Second DELOS Workshop - Interoperability and Metadata, Workshop Position Paper (considers Z39.50 and Java):

<http://www.area.pi.cnr.it/ErcimDL/second-DELOS-workshop/papers/Borbinha/delos.html>

(<http://www.area.pi.cnr.it/ErcimDL/second-DELOS-workshop/papers/Borbinha/delos.html>)

The Java Telnet Applet:

<http://www.first.gmd.de/persons/leo/java/Telnet/> (<http://www.first.gmd.de/persons/leo/java/Telnet/>)

Genny Engel (gengel@sfghdean.ucsf.edu (mailto:gengel@sfghdean.ucsf.edu)) is Information Services Librarian for San Francisco General Hospital and edits the BayCHI News, a monthly newsletter for members of the San Francisco Bay Area chapter of ACM SIGCHI (Special Interest Group in Computer-Human Interaction).

Copyright © 1997 by Genevieve Engel. 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 gengel@sfghdean.ucsf.edu (gengel@sfghdean.ucsf.edu).

table of contents



[ter issues \(/lita/publications/archive/ter\)](#)

About TER