

Volume 11, No 4, September 2004

**ter**  
Technology Electronic Reviews

---

Formerly *Telecommunications Electronic Reviews*

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

Technology Electronic Reviews (ISSN: 1533-9165) is a periodical copyright © 2004 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: Diane K. Kovacs and Kara L. Robinson. (2004). *The Kovacs Guide to Electronic Library Collection Development: Essential Core Subject Collections, Selection Criteria and Guidelines*. New York: Neal-Schuman. (ISBN: 1-55570-483-2) By Cindy Schofield-Bodt.
- REVIEW OF: Daniel Selman. (2002). *Java 3D Programming*. Greenwich, CT: Manning. (ISBN: 1-930110-35-9) By Ray Olszewski.
- REVIEW OF: Frederick Stielow. (2003). *Building digital archives, descriptions, and displays: a how-to-do-it manual for archivists and librarians*. New York: Neal-Schuman. (ISBN: 1-55570-463-8) By Bradford Lee Eden, Ph.D.
- REVIEW OF: Dejan Sunderic. (2003). *SQL Server 2000 Stored Procedure & XML Programming*. Emeryville, CA: McGraw-Hill/Osborne. (ISBN: 0-07222-896-2) By Yan Han.
- About TER

---

**REVIEW OF: Diane K. Kovacs and Kara L. Robinson. (2004). *The Kovacs Guide to Electronic Library Collection Development: Essential Core Subject Collections, Selection Criteria and Guidelines*. New York: Neal-Schuman.**

by Cindy Schofield-Bodt

Raganathan's Five Laws of Library Science are taught in all the intro classes--how exciting to see one Law's continuing relevance affirmed in the foreword of this book! When "reader" is modernized to include "web-surfer," the law "to save the time of the reader" guides Diane

Kovac's work as, she and co-author Kara Robinson provide a framework for "selectively collecting and making needed [electronic] information resources available." (p.xiii)

The guide is divided into three parts: Recommended Practices; Recommended Evaluation Guidelines, Selection Criteria, and Core Collections for Major Subject Areas; and Web Collection Development Resources. The introduction provides the reader with the address and password for a companion website where the core e-library resources are available in whatever revised and updated formats they've morphed into. (p. xvii)

Kovacs first tackled the topic of electronic collection development in 2000 with the publication of *Building Electronic Library Collections*. *The Kovacs Guide* is a necessary update that expands on the original work by continuing the coverage of the earlier title and adding information that reflects the evolving world of electronic resources. The most obvious developments in the new book address the trend to include Web-accessible resources in the library catalog, virtual reference and document delivery issues, the proliferation of World Wide Web fee-based databases, and growth in the availability of electronic journals and books. This time around "the authors have attempted to shape *The Kovacs Guide* as a 'one-stop' resource that tries to serve three distinct functions of electronic collection policies: to guide collection planning; to steer collection management; and to identify subject collection, criteria, and core electronic resources." (p.xv) The book is not a primer; it assumes that the reader has a familiarity with concepts of collection development and evaluation. It is a practical handbook that will become quickly dog-eared in the hands of any librarian attempting to cultivate an electronic library element in their resource ensemble.

"Part I: Recommended Practices" begins with the chapter "General Collection Development Principles for Web-Accessible Resources." It is arguably the part of the book that the librarian concerned with these issues should be curling up with and reading word-for-word. This is where the authors translate the collection development process to the Web environment. The authors make the convincing argument that the ubiquity of personal computers and the acceptance of the World Wide Web as a valid form of publication mandate that every type of library build Web-accessible resource collections. These resource collections, referred to as e-libraries throughout the book, are absolutely necessary if libraries are to continue to fulfill their service missions and remain relevant to those seeking information in their service communities.

Staffing choices need to be made hand-in-hand with the development of this new resource, and two models are presented. A new position may be created that is specifically devoted to electronic resources, or selection and acquisition of electronic resources may be incorporated into the workflow of a librarian's collection development responsibilities. (p.4) In either case, the library administration must provide training in at least two critical areas: selection criteria and resource accessibility. The authors build their collection development guide around the

premise that "electronic resources librarians can develop their collection of Web-accessible information resources...based on the same basic collection policies articulated by libraries for print and electronic resources accessed locally." (p.4)

While an "e-library vocabulary" is still developing, an informal survey of 700-plus librarians culled a thesaurus of words and phrases that allow in-depth discussion of electronic collection issues. Most respondents agreed that reference services offered through the Web are referred to as "virtual reference" and users of library services are "patrons." Librarians most often refer to a collection of Web-published, Web-accessible information resources as an "e-library" or "electronic library." An "Internet resource" refers to resources available through any number of Internet services (gopher, FTP, etc.) while "web resources" refers specifically to information published via the World Wide Web. The value of the vocabulary discussion is the confirmation that an agreed-upon vocabulary is not yet firm, and in the recognition that librarians and library patrons may use words quite differently is essential to clear communication. A useful tool for librarians is the Web publication "*Library Terms That Users Understand* (<http://www.jkup.net/terms.html>)" by John Kupersmith ([www.jkup.net/terms.html](http://www.jkup.net/terms.html)) (<http://www.jkup.net/terms.html>) that provides librarians with the most effective terms that users (patrons) understand, thereby enabling them to make productive choices.

In developing any new program or service, the first step is planning. The authors offer six important planning questions in the "Recommended Practices" section. Librarians are instructed to consider first and foremost whether an Internet connected computer with Web server software is available. Then, the need for other hardware or software purchases must be considered and budgeted, knowledgeable technical personnel need to be on board, reasonable collection development assignments (the right people, the right amount of time) need to be made and a commitment to training must be demonstrated.(p.12)

A checklist is also provided for use in the development of a collection plan. The twin questions "What purpose will your e-library serve?" and "Why do you want to build an e-library" (p.16) should be the first consideration. Following that discussion the plan needs to be further refined with subject areas to be included, the basis for subject focus, distance learner issues, intended audience, inclusion of free or fee-based resources, organizational strategy, and finally, types of related virtual reference service to be provided. With the carefully considered plan in place, collection of resources can begin.

The sections "Identifying and Collecting Web Resources" and "General Evaluation Guidelines" continue the pattern of listing and then elaborating on concepts. The text here is filled with citations for lists of discussion groups, newsgroups, website reviews, e-journals and e-newsletters that evaluate Web resources, Web logs that publish reviews and print books and journals that review Web resources. Evaluation of sources is dependant on a clear understanding of the nature of Internet information, awareness of the basic problems of Internet information, and the ability to determine the source of Web published information. Each area is discussed in detail.

The final bit of wisdom shared in chapter 1 has to do with maintaining the Web accessible e-library. Again, compared to traditional collection development, "building a Web-based e-library is an ongoing process." (p.29) Weeding continues to be an on-going chore, and link checking is akin to "shelf-reading" in the book stacks. The authors argue that an important element of any collection is a mechanism for patron feedback. "Mailto" links will work, but since they often attract spammers, the authors have included the Perl CGI script used on the book's companion website for reproduction by interested readers.

Chapter 2 deals with the "hands-on" issues of providing the same service, support, and access for the e-library collection as is provided for the paper library. The three areas examined are reference service, cataloging, and ongoing creation of digital collections. Various options for providing reference service are outlined, and strategies for training staff in virtual reference techniques are presented. A comprehensive overview of descriptive cataloging options for Web resources is preceded by a succinct argument in favor of including web resources in the library catalog. Both Sanford Berman and cataloging manager at Kent State University Margaret Maurer weigh in with their opinions on the importance of providing catalog records with multiple access points for these new forms of information. (p.50-51) A section on maintaining and developing digital collections warns of the labor-intensive and complex nature of the task, but is generously illustrated with the "success stories" trademark of this guide.

The final chapter of Part I covers licensing basics. A relatively short introduction including a review of recent literature, key terms and ample references, the chapter outlines important license issues and provides a framework from which to further explore the licensing issues faced when considering the inclusion of certain Web resources.

Parts II and III will cause this book to be reached for again and again as librarians use these recommendations and resources in their hands-on collection development tasks.

Recommended Evaluation Guidelines, Selection Criteria, and Core Collections for Major Subject Areas include eight points for each nine subject areas. The nine sections of information resources are: Ready-Reference, Business, Jobs & Employment, Medical, Legal, Biological Sciences, Physical Sciences, Social Sciences & Education, and Reader's Advisory & Collection Development. A librarian with interest in only one of these areas will benefit greatly from reading Part I and then choosing their area of interest from Part II.

Part III will be useful to all readers as it is an assemblage of resources organized into three lists: 1. Collection Development Related Discussion Lists, Newsgroups and Blogs. 2. Evaluation Guides. 3. Multiple Subject Resource Collection Tools.

Kovacs and Robinson have succeeded and in writing a guide that is both readable and useful. The "E-Library Success Stories" that are included in each chapter enhance the book by providing outlines of library successes that serve as inspiration and idea generators. The comprehensive index provides adequate "back-door" access to the content. While it is extremely difficult to write a book about library technology issues that will remain relevant for

longer than a high school lunch period, this work will stay in circulation for a while. The authors are applying standard collection development concepts to the new electronic format, not re-inventing the wheel. After one reads the book, it all seems so obvious.

*Cindy Schofield-Bodt, Librarian, Southern Connecticut State University, is currently the Buley Library Technical Services Division Head. She has taught cataloging and acquisitions courses as an adjunct at the SCSU Graduate School of Library And Information Science.*

Copyright © 2004 by Cindy Schofield-Bodt. 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 [schofieldbc1@southernct.edu](mailto:schofieldbc1@southernct.edu) (<mailto:Schofieldbc1@southernct.edu>) .

---

## **REVIEW OF: Daniel Selman. (2002). *Java 3D Programming*. Greenwich, CT: Manning.**

by Ray Olszewski

New languages for computer programming are introduced to the profession with a clockwork-like regularity. Most either fade into obscurity (Ada, Modula-2, even Pascal), while some find a role serving specialized niches (Perl, Python). Only rarely does a new language join the small group of major programming languages.

Java, first introduced by Sun Microsystems a bit less than a decade ago, has become one of these rare, breakout languages. Versions of it have been adapted to a wide variety of uses, including low-level embedded systems, Web-page applets, and native-code applications for Wintel and Macintosh PCs. Written as a very high level language, a big part of its strength is that it incorporates standard libraries that allow developers to use high level abstractions to describe the objects and behaviors they want their programs to represent.

Java 3D Programming is a guide to using a particular high-level library to create and operate views into three-dimensional scenes on a computer screen. Specifically, it explains use of the Applications Programming Interface (API) embodied by the Java 3D SDK (Software Development Kit). This API provides classes -- a class is, in essence, a way to define an object with particular properties and behaviors; an example will come shortly -- that let programmers create three dimensional spaces, create three dimensional objects, locate the objects in space, and cause them to do things like move, change shape and size, and collide.

For example, one use of this API is to create a simple object like a cube. A cube has many properties, including size, color, position in 3D space (x, y, and z coordinates) relative to a grid, rotation relative to each axis of that grid (conventionally called pitch, roll, and yaw), and surface texture. It can also do things, such as change size, rotate relative to the axes, and move relative to the grid. Standard programming languages (including Java, without this API) require the programmer to create code to handle all these behaviors explicitly. This API simplifies the process by predefining characteristics and behaviors of standard objects, as well as ways that objects can be used to construct more complex objects.

Java 3D Programming does an excellent job of describing the properties and methods associated with the main objects available in the API. The bulk of the book is a systematic exploration of the API, covering creation of the background "universe"; primitive objects such as spheres, boxes, and cylinders; lighting and surface texture; the basic behaviors of objects; and interaction with the user through a mouse and keyboard. The author also offers some help in getting familiar with the construction of complex objects from simple ones, and with more advanced behaviors like detecting and responding to collisions.

As with just about any programming text, the example programs that accompany this one (available via a Web-based download) are a big part of the instruction. For many programmers (me included), examples tell most of the story. For the most part, Selman does a good job on this front, providing a large assortment of examples that illustrate most of the aspects of 3D programming that the text discusses in more general terms. Even better, he is good at writing example code, combining a clear programming style with a disciplined use of comments to explain his intent.

For the experienced Java programmer, this book provides an excellent way to get quickly up to speed with the Java 3D API.

The only real flaw in the book is that the publisher tries to oversell it, claiming far more than what the author delivers. The back-cover blurb asserts that "Novice programmers will gain fast entry into Java 3D development". This is pure nonsense. The book assumes far too much familiarity with the conventions of Java to be even minimally comprehensible to a true novice. Although it includes a token introduction to 3D concepts, the treatment is minimal and cursory; only the list of other references to consult offers a beginner any shot at understanding the basics. Even the instructions for installation of the Java SDK and the book's own examples are flawed in ways that would flummox a beginner.

Other weak points are less weaknesses of the book than of Java and the 3D API themselves. Java demands a lot of computer resources, requiring a fast CPU and a lot of RAM. A very high level API like the 3D one increases this demand; seemingly simple instructions to the API can create a lot of work under the hood, increasing this requirement. My test system, a 1.7 GHz Celeron with 1024 MB RAM, had trouble running some of the examples that involved fast

motion. Selman is careful to point out this limitation of Java and the API, but it deserves greater emphasis and, perhaps, more detailed treatment (and discussion of workarounds) than he gives it.

Although Java is in conception a platform-independent language -- the same code is supposed to run the same way on, for example, Windows, Macintosh, and Linux systems -- in practice it falls short of this ideal. In the case of the 3D API, two (Windows) versions are available, using different low-level graphics commands. The version Selman recommends (the OpenGL version) simply did not work on my test system -- it would not render spheres correctly, using either Selman's example code or the API's own test code. I had to switch to the DirectX version of the API even to run the examples. In this instance, Selman's installation instructions were of no help at all, since he limits his discussion to a blanket recommendation of the OpenGL version as "more stable and ahead of the DirectX release in terms of features" (page 28).

So, if you know core Java and want to get up to speed quickly on its 3D API, this is the book for you. But if you do not know Java well, you would do better to look elsewhere for a more basic instruction, until you are sufficiently experienced to assimilate a high-speed introduction to top end Java APIs like the one this book discusses.

*Ray Olszewski is a Senior Software Engineer at Protogene Laboratories, where he focuses on embedded systems development. He has also worked as a freelance computer programmer and statistician. His work includes development of custom Web-based software to support on-line research.*

Copyright © 2004 by Ray Olszewski. 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 [ray@comarre.com](mailto:ray@comarre.com) (<mailto:ray@comarre.com>) .

---

**REVIEW OF: Frederick Stielow. (2003). *Building digital archives, descriptions, and displays: a how-to-do-it manual for archivists and librarians*. New York: Neal-Schuman.**

by Bradford Lee Eden, Ph.D.

The intended audience for this book is professionals in libraries, archives, and museums; those in cultural and historical associations; and students. The author is a librarian and archivist whose work in the National Park Service and museum community is well-known. Mr. Stielow has authored and/or edited ten books, including the 1999 *Creating a Virtual Library* in

Neal-Schuman's How-to series. In the preface, the author indicates that this book is geared primarily towards archivists, then librarians, and finally museums and historical agencies. He defines what a "digital archive" is, indicates that this book is a primer for project management, and eventually evolves into a general guide for managing digital archives and projects.

Chapter 1, "Planning the digitization project," provides frameworks for initial planning, including systems analysis, setup, and electronic records management. A technological overview of digital concepts such as the Internet, XML, RDF, HTML, and other terms are explained. "Understanding digitized finding aids," Chapter 2, is primarily a training exercise in automation project management and digital archival finding aids. Chapter 3, "Creating effective web finding aids," introduces HTML and hypertext tool skills to convert print finding aids into digital ones. In "Considering SGML/EAD, XML, and database options," the author explains some of the more structured and technologically-advanced database options available.

In Chapter 5, "Exploring leading automation systems for libraries and museums," legacy automation systems are discussed and examined, specifically online bibliographic utilities, Dublin Core, Open URL, established descriptive standards, and commercial turnkey applications. Concerns about copyright, policy issues, and selection criteria for in-house digitization and Web presentation projects are covered in Chapter 6, "Establishing policies and techniques for digital imaging." In Chapter 7, "Creating the web site and display options," the author provides a top-down view of a model digital archives through four manageable layers: a formal entrance or gateway, displays, descriptive devices, and digital assets. Finally, in Chapter 8, "Maintaining and preserving digital archives," methods for preserving and maintaining digital archives in three institutional situations (web museums, institutional records, and digital libraries) are presented. A short afterword titled "Setting the historical context" is a conclusion to help readers tie everything together. A webliography and index supplement the text.

One of the wonderful things about the How-to series is that it presents technological and challenging material in a simple and straightforward way to the uninitiated and the beginning learner. This book brings together all of the important materials available regarding starting, planning, directing, digitizing, and presenting digital materials on the Internet, as well as practical advice, options, and alternatives for those with few resources or plenty of resources to work with. Colored boxes throughout the text highlight personal and practical comments from the author. Overall, I would highly recommend this book to archivists, librarians, and other information organization professionals who are involved in web, digitization, and digital projects in their institutions.

*Brad Eden, Ph.D., Head, Web and Digitization Services, University of Nevada, Las Vegas.*

Copyright © 2004 by Bradford Lee Eden, Ph.D. 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) (mailto:[beden@ccmail.nevada.edu](mailto:beden@ccmail.nevada.edu)).

---

**REVIEW OF: Dejan Sunderic. (2003). *SQL Server 2000 Stored Procedure & XML Programming*. Emeryville, CA: McGraw-Hill/Osborne.**

by Yan Han.

"Stored procedure" is defined as "a precompiled collection of SQL statements and optional control-of-flow statements stored under a name and processed as a unit...". (Microsoft, 2002, 499) *SQL Server 2000 Stored Procedure & XML Programming* identifies key concepts and best practices to use stored procedures and XML programming in the SQL Server 2000 environment. The intended audiences are novice and intermediate SQL Server developers. The previous version of the book was published in 2001 with the title *SQL Server 2000 Stored Procedure Programming*. It is clear that the author updated some topics and introduced a new section, programming with XML.

The book consists of 15 chapters arranged in two parts, along with an appendix. The first part, comprised of the first 12 chapters, covers stored procedure programming in the SQL Server 2000 environment. The second part, chapters 13 to 15, introduces the basic concept of XML, XML Schema, and XML interaction with SQL Server 2000. In addition, a sample database is available for downloading from the author's website and notes on the installation are also provided.

Each chapter describes one conceptual topic as well as related practices. Chapter 1 is the introduction to the contents of the book and procedures to set up the SQL Server development environment. Chapter 2 gives an overview of the SQL Server 2000 environment, including the SQL Server 2000 tools (e.g. Service Manager, Query Analyzer, Enterprise Manager) and basic operations on stored procedures (e.g. execution, editing). Other chapters are dedicated to concepts and best practices of using stored procedures, including design concepts, the Transact-SQL programming language, Transact-SQL constructs, built-in functions, and interaction with the SQL Server environment. General best practices on topics such as naming conventions (chapter 1), debugging, and error handling (chapter 7) are critical for both novice and intermediate developers. Intermediate readers may also find chapter 12 (stored procedures for web search engines) of particular interest.

The book, published in 2003, is current and well written. The author is a principal consultant in an IT consulting company and published the first version of this book in 2001. His writing style is clear and concise, as the book is clearly written and includes many examples. These are not complex and are usually accompanied by screen captures, which adequately explain related

concepts or topics. The introduction to SQL Server 2000 and the procedure to set up the database are very useful for novice users to quickly get into the topics. The book is very well organized, with a detailed table of contents and a thorough index. It is printed on quality paper with regular fonts and font sizes. In addition, different contents or concepts can be easily distinguished because multiple fonts and sizes are used for easy recognition.

However, readers may find that the introductions to XML and XML Schema are not adequate. The author obviously missed the concept of "prolog", a component of XML, in the topic. The introduction to XML Schema is limited in scope with only eleven pages, ignoring some advanced concepts (e.g. datatypes, structures). In addition, due to rapid development in the XML field, some information on XML Schema is outdated. For instance, the MSXML parser now supports the W3C final recommendation for XML Schema. The official XML and XML Schema standards on the web (<http://www.w3.org/XML/> (<http://www.w3.org/XML/>) and <http://www.w3.org/XML/Schema> (<http://www.w3.org/XML/Schema>)) are highly recommended for reading and reference.

Finally, readers should be aware of an error in *Microsoft Computer Dictionary* as well as a commonly held misperception, that stored procedures are pre-compiled. Stored procedures are not pre-compiled or partially pre-compiled. According to Microsoft's description: "SQL Server 2000 and SQL Server version 7.0 incorporate a number of changes to statement processing that extend many of the performance benefits of stored procedures to all SQL statements. SQL Server 2000 and SQL Server 7.0 do not save a partially compiled plan for stored procedures when they are created. A stored procedure is compiled at execution time, like any other Transact-SQL statement. SQL Server 2000 and SQL Server 7.0 retain execution plans for all SQL statements in the procedure cache, not just stored procedure execution plans."(Microsoft, 2004)

The book is a great starting point for novice readers to explore the world of stored procedures and XML programming in the SQL Server environment. Its coverage on stored procedures is adequate and is highly recommended for novice and intermediate readers who have interests in the topic of stored procedures SQL Server 2000 environment. It is a good computer science book for public and academic libraries' collections.

Reference:

[1] *SQL server books online*. (2004). Redmond Washington: Microsoft. Available with the SQL Server 2000

[2] *Microsoft computer dictionary* (5th ed.). (2002). Microsoft Press. Redmond, Wash.: Microsoft Press.

*Yan Han, Systems Librarian, University of Arizona Library.*

Copyright © 2004 by Yan Han. 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 hany@u.library.arizona.edu (mailto:hany@u.library.arizona.edu).

---

## **About TER**

Editor is Sharon Rankin, McGill University ( sharon.rankin@mcgill.ca (mailto:sharon.rankin@mcgill.ca)). Editorial Board Members are: Linda Robinson Barr, Texas Lutheran University ( lbarr@tlu.edu (mailto:lbarr@tlu.edu)); Paul J. Bracke, Arizona Health Sciences Library ( paul@ahsl.arizona.edu (mailto:paul@ahsl.arizona.edu)); Kathlene Hanson, California State University, Monterey Bay ( kathlene\_hanson@csumb.edu (mailto:kathlene\_hanson@csumb.edu)); Adriene Lim, Wayne State University ( ab7155@wayne.edu (mailto:ab7155@wayne.edu)); Tierney Morse McGill, Colorado State University ( tmcgill@manta.colostate.edu (mailto:tmcgill@manta.colostate.edu)); Florence Tang, Mercer University, Atlanta ( tang\_fy@mercer.edu (mailto:tang\_fy@mercer.edu)); Stacey Voeller, Minnesota State University ( voeller@mnstate.edu (mailto:voeller@mnstate.edu)); and Laura Wrubel, University of Maryland ( lwrubel@umd.edu (mailto:lwrubel@umd.edu)); Michael Yunkin, University of Nevada, Las Vegas ( michael.yunkin@ccmail.nevada.edu (mailto:michael.yunkin@ccmail.nevada.edu)).

---