
DOWNLOAD FULL VERSION HERE

Graphics Programming With Gdi+, Mahesh Chand, 2004, Computers, 740 pages. & All Windows programmers developing applications that deal with graphics, monitors, or printers need to use GDI+. & & There is little documentation available on GDI+. There ....


Schaum's Outline of Computer Graphics 2/E, Zhigang Xiang, Roy A. Plastock, Sep 8, 2000, Computers, 347 pages. Scores of examples and problems allow students to hone their skills. Clear explanations of fundamental tasks facilitate students understanding of important concepts. New ....


Photoshop CS Bible, Deke McClelland, Mar 5, 2004, Computers, 1072 pages. "I've learned more from Deke's Photoshop Bible than any Photoshop book, ever." Scott Kelby, President, National Association of Photoshop Professionals "When someone asks me ....

Beginning OpenGL Game Programming, Dave Astle, Kevin H. Hawkins, 2004, Computers, 310 pages. Discusses the basics of using OpenGL to create computer games that have realistic graphics.


OpenGL Game Programming, Kevin Hawkins, Dave Astle, 2001, Computers, 777 pages. OpenGL Game Programming teaches you how to use OpenGL to create dynamic 3D environments and effects for use in game development. Beginning with the histories of OpenGL and ....
The OpenGL Extensions Guide, Eric Lengyel, 2003, Computers, 670 pages. Until now, the game industry was lacking one single, concise reference to help make sense of the dozens of extensions available. "The OpenGL Extensions Guide" provides this ....

Learn 3D graphics programming on the PC, Richard F. Ferraro, 1996, Computers, 1022 pages. Offering step-by-step instructions on how to create 3D graphics applications, a guide for programmers at any level also provides a commercial 3D graphics library on the ....


OpenGL programming for the X Window System, Mark J. Kilgard, 1996, Computers, 542 pages. SGI's X Windows graphics expert explains how to construct real and useful 3D applications using OpenGL and X, and how to tightly integrate OpenGL applications with the X Window ....


OFFICIAL OPENGL LIBRARY, OpenGL Architecture Review Board, Jackie Neider, Mason Woo, Tom Davis, Dave Shreiner, Apr 1, 2004, , 1184 pages. A value-priced boxed gift set of the two key books on OpenGL...

OpenGL SuperBible, Richard S. Wright, Michael Sweet, 2000, , 696 pages. Offering guidance in applying the complex concepts and details, "OpenGL SuperBible" serves both as a reference and as a tutorial manual that covers everything the readers needs ....

New! Introducing the tech.book(store), a hub for Software Developers and Architects, Networking Administrators, TPMs, and other technology professionals to find highly-rated and highly-relevant career resources. Shop books on programming and big data, or read this week's blog posts by authors and thought-leaders in the tech industry. > Shop now

The Library enables programmers to gain a practical understanding of OpenGL and shows them how to unlock its full potential. Originally developed by SGI, the Library continues to evolve under the auspices of the Architecture Review Board (ARB), an industry consortium responsible for guiding the evolution of OpenGL and related technologies. The OpenGL ARB is composed of leaders in the computer graphics industry: 3Dlabs, Apple, ATI, Dell, Evans & Sutherland, Hewlett-Packard, IBM, Intel, Matrox, NVIDIA, SGI, and Sun Microsystems.

First introduced in 1992, OpenGL is an industry standard graphical application programming interface (API) that supports 2D and 3D rendering across a host of platforms. The Architectural Review Board (ARB) governs the OpenGL API and oversees the adoption of new interface functions. Functions (or commands) within the API are usually simple and discrete. A developer calls a series of these small functions in sequence to specify rendering operations. To help utilize the library, the "OpenGL Reference Manual" supplies key functional documentation in a uniform manner.

The third and fourth chapters list different groupings of the functional commands to provide the reader with several methods to index and reference functions. The third chapter details all each official OpenGL command categorized by functionality. The fourth chapter lists the various OpenGL constants that are compatible with each command.

Beginning with the fifth chapter, 160 official OpenGL commands are described. Listed alphabetically, every command has the following sections: Name, Function Prototype, Parameters, Description, Notes, Errors, See Also, and (sometimes when appropriate) Associated Gets. The coverage of each command spans an average of 3 pages.
Overall, the organization and consistency is excellent. Often, material is duplicated per command to save the reader cross-referencing other sections of the book. Throughout the text, the wording is clear and unambiguous (if a bit dry) - exactly what you'd expect from a reference book of this nature.

The book does have a few shortcomings, however. There is only a small trace of sample source code. While the commands are presented alphabetically by class, the book contained no overall index. OpenGL Extensions (pixel and vertex shader commands, etc.) are not provided since they're not officially part of the Standard. Finally, having an electronic version of the text would have been a nice touch - especially one that integrated with the common development environments to provide context sensitive help or electronic searching.

The latest edition of the "OpenGL Reference Manual" is a great companion for OpenGL developers. To get the most from this book, readers unfamiliar or interested in learning the API should first read the "OpenGL Programming Guide, 4th Edition" (ISBN 0-3-211-73491) also published by Addison Wesley. Read more &rsaquo;

OpenGL is a powerful software interface used to produce high-quality computer generated images and interactive graphics applications by rendering 2D and 3D geometric objects, bitmaps, and color images. Officially sanctioned by the OpenGL Architecture Review Board (ARB), the OpenGL® Reference Manual, Third Edition, is the comprehensive and definitive documentation of all OpenGL functions. This third edition covers OpenGL Version 1.2, including its newest features: 3D texture mapping; multitexturing; mipmapped texture level-of-detail control; new pixel storage formats, including packed and reversed (BGRA) formats; rescaling vertex normals; and specular lighting after texturing. In addition, this book documents the newest routines in the OpenGL Utility Library (GLU 1.3) and added functionality in the OpenGL extension to the X Window System (GLX 1.3). The comprehensive reference section documents each set of related OpenGL commands. Each reference page contains: * A description of the command's parameters * The effects on rendering and the OpenGL state by the command * Examples * Errors generated by functions * References to related functions This book also includes a conceptual overview of OpenGL, a summary of commands and routines, a chapter on defined constants and associated commands, and a description of the ARB extensions, including multitexture and the imaging subset. The OpenGL Technical Library provides tutorial and reference books for OpenGL. The Library enables programmers to gain a practical understanding of OpenGL and shows them how to unlock its full potential. Originally developed by SGI, the Library continues to evolve under the auspices of the Architecture Review Board (ARB), an industry consortium responsible for guiding the evolution of OpenGL and related technologies. The OpenGL ARB is composed of industry leaders, such as 3Dlabs, Compaq, Evans & Sutherland, Hewlett-Packard, IBM, Intel, Intergraph, Microsoft, NVIDIA, and SGI. The OpenGL® Reference Manual, Third Edition, has been completely revised and updated for OpenGL, Version 1.2, by Dave Shreiner, in collaboration with the ARB.0201657651B04062001

Officially sanctioned by the OpenGL Architecture Review Board (ARB), the OpenGL® Reference Manual, Third Edition, is the comprehensive and definitive documentation of all OpenGL functions. This third edition covers OpenGL Version 1.2, including its newest features: 3D texture mapping; multitexturing; mipmapped texture level-of-detail control; new pixel storage formats, including packed and reversed (BGRA) formats; rescaling vertex normals; and specular lighting after texturing. In addition, this book documents the newest routines in the OpenGL Utility Library (GLU 1.3) and added functionality in the OpenGL extension to the X Window System (GLX 1.3).

It is vital that you have this book when programming in OpenGL, especially when learning and you have not yet memorized even the more common prototypes. This book is NOT a replacement for the OpenGL Programming Guide, it is meant to accompany it and be a more useful reference once you have read the Programming Guide. Very VERY useful. Well Organized, important information jumps right out at you and details are available on the same page. Overall one of the best references I've used for any language or API.
This book goes into great detail on OpenGL, with some very helpful examples and lots of code fragments. All the examples from the book can be found on the enclosed CD. A little too complex for people who are new to C++ though. I found myself struggling with this book at first but, when my grasp of the language became better, this became a gold mine of information.

OpenGL is a powerful software interface used to produce high-quality, computer-generated images and interactive applications using 2D and 3D objects, bitmaps, and color images. The OpenGL (R) Programming Guide, Fifth Edition, provides definitive and comprehensive information on OpenGL and the OpenGL Utility Library. The previous edition covered OpenGL through Version 1.4. This fifth edition of the best-selling "red book" describes the latest features of OpenGL Versions 1.5 and 2.0, including the introduction of the OpenGL Shading Language. You will find clear explanations of OpenGL functionality and many basic computer graphics techniques, such as building and rendering 3D models; interactively viewing objects from different perspective points; and using shading, lighting, and texturing effects for greater realism. In addition, this book provides in-depth coverage of advanced techniques, including texture mapping, antialiasing, fog and atmospheric effects, NURBS, image processing, and more. The text also explores other key topics such as enhancing performance, OpenGL extensions, and cross-platform techniques. This fifth edition has been extensively updated to include the newest features of OpenGL Versions 1.5 and 2.0, including:

- Storage of vertex arrays in buffer objects for faster rendering
- Occlusion queries for course-grain visibility testing
- Non-power-of-two dimensioned texture maps
- Point sprites
- Separate stencil operations for RGB and alpha
- Rendering to multiple color buffers using GLSL

Most importantly, this edition discusses the OpenGL Shading Language (GLSL) and explains the mechanics of using this new language to create complex graphics effects and boost the computational power of OpenGL.

The OpenGL Programming Guide, now in its third edition, is the definitive volume for programmers using this evolving graphics interface standard. Written by members of the OpenGL Architecture Review Board, this book offers understandable tutorials and lessons on getting up to speed and getting the most out of the latest version of OpenGL, version 1.2.

The guide uses code examples in C and is targeted at programmers who have experience in coding yet are new to coding for OpenGL applications. The opening chapters go into descriptive detail of how OpenGL, the software interface for hardware 3-D chipsets, works and what you can expect from it, which turns out to be much more than you might have thought. Color plates are used, for example, to show how OpenGL handles such effects as motion blur and depth-of-field blur, in addition to shadows and texture mapping.

This is not a beginner's guide to programming computer graphics. Some previous knowledge of both programming in general and computer graphics in particular is required. For example, code snippets are used to describe how to implement these effects, but because OpenGL is platform-independent, some code examples may need to be modified when used with your specific compiler.

Filled with the expertise of those who standardized OpenGL, there is no better reference volume for learning and understanding this system. The examples cited are clear, commented, and explained. The only drawback to the book is that it lacks a companion CD-ROM—all examples must be either typed in or downloaded from an Internet FTP site. (The URL is listed in the preface.) --Mike Caputo

--This text refers to an out of print or unavailable edition of this title.

The OpenGL graphics system is a software interface to graphics hardware. (The GL stands for Graphics Library.) It allows you to create interactive programs that produce color images of moving three-dimensional objects. With OpenGL, you can control computer-graphics technology to produce realistic pictures or ones that depart from reality in imaginative ways. This guide explains how to program with the OpenGL graphics system to deliver the visual effect you want.

Chapter 5, "Lighting," explains how to control the lighting conditions surrounding an object and how that object responds to light (that is, how it reflects or absorbs light). Lighting is an important topic, since objects usually don't look three-dimensional until they're lit. The remaining chapters explain
how to optimize or add sophisticated features to your three-dimensional scene. You might choose not to take advantage of many of these features until you're more comfortable with OpenGL. Particularly advanced topics are noted in the text where they occur.

Chapter 6, "Blending, Antialiasing, Fog, and Polygon Offset," describes techniques essential to creating a realistic scene--alpha blending (to create transparent objects), antialiasing (to eliminate jagged edges), atmospheric effects (to simulate fog or smog), and polygon offset (to remove visual artifacts when highlighting the edges of filled polygons).

Chapter 13, "Selection and Feedback," explains how you can use OpenGL's selection mechanism to select an object on the screen. Additionally, the chapter also explains the feedback mechanism, which allows you to collect the drawing information OpenGL produces rather than having it be used to draw on the screen.

Chapter 14, "Now That You Know," describes how to use OpenGL in several clever and unexpected ways to produce interesting results. These techniques are drawn from years of experience with both OpenGL and the technological precursor to OpenGL, the Silicon Graphics IRIS Graphics Library. In addition, there are several appendices that you will likely find useful.

Appendix C, "OpenGL and Window Systems," briefly describes the routines available in window-system-specific libraries, which are extended to support OpenGL rendering. Window system interfaces to the X Window System, Apple MacIntosh, IBM OS/2, and Microsoft Windows NT and Windows 95 are discussed here.

This guide assumes only that you know how to program in the C language and that you have some background in mathematics (geometry, trigonometry, linear algebra, calculus, and differential geometry). Even if you have little or no experience with computer-graphics technology, you should be able to follow most of the discussions in this book. Of course, computer graphics is a huge subject, so you may want to enrich your learning experience with supplemental reading.

Once you begin programming with OpenGL, you might want to obtain the OpenGL Reference Manual by the OpenGL Architecture Review Board (Reading, MA: Addison-Wesley), which is designed as a companion volume to this guide. The Reference Manual provides a technical view of how OpenGL operates on data that describes a geometric object or an image to produce an image on the screen. It also contains full descriptions of each set of related OpenGL commands--the parameters used by the commands, the default values for those parameters, and what the commands accomplish. Many OpenGL implementations have this same material on-line, in the form of manual pages or other help documents, which are probably more up-to-date. There are also versions on the World Wide Web; consult the previously mentioned Official OpenGL Web Site.

OpenGL is really a hardware-independent specification of a programming interface, and you use a particular implementation of it on a particular kind of hardware. This guide explains how to program with any OpenGL implementation. However, since implementations may vary slightly--in performance and in providing additional, optional features, for example--you might want to investigate whether supplementary documentation is available for the particular implementation you're using. In addition, you might have OpenGL-related utilities, toolkits, programming and debugging support, widgets, sample programs, and demos available to you with your system.

This guide contains many sample programs to illustrate the use of particular OpenGL programming techniques. These programs make use of Mark Kilgard's OpenGL Utility Toolkit (GLUT). GLUT is documented in OpenGL Programming for the X Window System by Mark Kilgard (Reading, MA: Addison-Wesley Developers Press, 1996). The section "OpenGL-Related Libraries" in Chapter 1
and Appendix D give more information about using GLUT. If you have access to the Internet, you
can obtain the source code for both the sample programs and GLUT for free via anonymous ftp
(file-transfer protocol).

Many implementations of OpenGL might also include the code samples as part of the system. This
source code is probably the best source for your implementation, because it might have been
optimized for your system. Read your machine-specific OpenGL documentation to see where the
code samples can be found.

Nate Robins has written a suite of tutorial programs that demonstrate basic OpenGL programming
concepts by allowing the user to modify the parameters of a function and interactively see their
effects. Topics covered include transformations, lighting, fog, and texturing. These highly
recommended tutorials are portable and require the aforementioned GLUT. To get the source code
for these tutorials, see this Web site: cs.utah/~narobins/opengl.html.

These style conventions are used in this guide: Bold--Command and routine names and matrices
Italics--Variables, arguments, parameter names, spatial dimensions, matrix components, and first
occurrences of key terms Regular--Enumerated types and defined constants Code examples are set
off from the text in a monospace font, and command summaries are shaded with gray boxes.

In a command summary, braces are used to identify choices among data types. In the following
example, glCommand has four possible suffixes: s, i, f, and d, which stand for the data types
GLshort, GLint, GLfloat, and GLdouble. In the function prototype for glCommand, TYPE is a
wildcard that represents the data type indicated by the suffix:

The 3rd Edition of the "OpenGL Programming Guide" is an important upgrade to what is the
definitive introduction to OpenGL programming. I was pleased to participate in the technical review
of this book so I can attest to (and I guess be held indirectly accountable for) the book's
completeness and accuracy.

OpenGL 1.2 is packed with new features like volume textures, image processing capabilities, more
image formats, etc, etc. The book covers _all_ the new stuff in OpenGL 1.2. OpenGL multitexturing
is already widely available. Games such as the much anticipated "Quake III: Arena" use OpenGL
multitexturing, and I expect lots of other 3D games will be using multitexturing as well.

Since the book uses the OpenGL Utility Toolkit (GLUT), all the examples can be compiled and run
on basically all OpenGL implementations, independent of operating system (Linux, Windows
95/NT, IRIX, MacOS, etc). The updated book also contains appendices that detail operating system
specific OpenGL usage.

But I bet most OpenGL programmers already have an earlier edition of this book so the big question
is whether the new edition is worth it. If you still have the 1st edition, getting the 3rd is a no-brainer.
If you have the 2nd edition, the main benefit of the 3rd edition is the new coverage of OpenGL 1.2
and multitexturing and the improved coverage of operating system support.

For those graphic programmers who think that Java3D requires that you program at too high a level
and want more elementary control over their graphics operations, OpenGL is probably a good
choice for a graphics programming API. So, assuming you already have a good grasp of computer
graphics programming concepts and can already program in the C language, this guide is probably
your best bet. If you don't already know these things, you will be completely lost trying to read this
book. The book starts out in chapter one showing the reader exactly what OpenGL can and cannot
do. Supporting code and instructive figures are supplied. The next few chapters deal with performing
basic graphics tasks by using the OpenGL API. Topics included are viewing, display lists, color,
lighting, blending, antialiasing, and fog. Again, if you are not already familiar with how to write
pseudocode for these type of graphics tasks, you need to find a good basic computer graphics text
and learn the techniques first. The next part of the book moves on to slightly more advanced topics
such as texture mapping, tessellators, quadrics, and NURBs all from the standpoint of "How do I do
this in OpenGL?”. The chapter entitled "Now That You Know" is especially valuable to the experienced graphics programmer. Unlike previous chapters, this chapter presents a collection of examples of higher-level graphics capabilities and some incomplete OpenGL code for those examples. The reader is expected to flesh out the code to get a running graphics program, and the book does point this fact out. Chapter 15, "OpenGL 20 and the OpenGL Shading Language", is the only truly new material in this 5th edition of the OpenGL programming guide. Thus, if you are not interested in using the OpenGL shader, you can probably forge ahead with the 4th edition of the guide if you already have a copy.

For those readers who want to learn OpenGL but have not yet learned computer graphics techniques, I highly recommend "Computer Graphics Using Open GL" (2nd Edition) by Francis Hill. That book has all of the basic graphics techniques explained using OpenGL code. However, note that it is a computer graphics textbook, NOT an OpenGL tutorial. Also, you might want to pick up a copy of Schaum's outline of computer graphics. It is an inexpensive guide to both 2D and 3D computer graphics techniques that is not specifically aimed at any particular language or API, and it has many excellent exercises that help you learn the material. Read more &rsquo;

The OpenGL Programming Guide has become a standard to which other 3D programming books are compared. It serves two basic functions: a tutorial of topical 3D concepts inherent in todays 3D graphics hardware/software and an instructional manual offering a precise explanation of the functional calls with supporting attributes/arguments defined by the OpenGL Application Programming Interface. The book is extremely well organized and allows readers to focus on specific topics while still maintaining perspective of the entire rendering pipeline. It is comprehensive, up-to-date and easy to read making it my first choice for clarifying all 3D technical issues. I cannot imagine a serious 3D programmer or software architect not having this book included in their personal library.

This book is a must for those wanting to fully exploit the power of OpenGL. In this 3rd Edition, the authors have included the newest features of OpenGL, Version 1.2. In particular, the discussion about multitexturing and imaging subset deserves a special mention, since they represent the latest rendering techniques available in form of a standard library.

Although all features, from the most basic to the most advanced technique, are covered in this book, it is not an introductory graphics textbook. Readers should have a basic knowledge about computer graphics in order to benefit from this book. Also, it is useful to be familiar with 2D/3D geometry and other graphics libraries. As code fragments and samples are written in C, experience with this programming language is also required.

OpenGL is a software interface to graphics hardware (the "GL" stands for Graphics Library). This interface consists of several hundred functions that allow graphics programmers to specify the objects and operations needed to produce high-quality color images of three-dimensional objects. Many of the functions are actually simple variations of each other, so in reality there are only about 190 substantially different functions.

Chapter 5, OpenGL Reference Pages, which forms the bulk of this manual, contains reference pages describing each set of related OpenGL commands. Commands with parameters that differ only in data type are described together. Each reference page describes the parameters, the effect of the commands, and what errors might result from using the commands.

What You Should Know Before Reading This Manual
This manual is intended as the companion reference volume to the fourth edition of the OpenGL Programming Guide by the OpenGL Architecture Review Board, Dave Shreiner, Mason Woo, Jackie Neider, and Tom Davis (Boston, MA: Addison-Wesley, 2003). Both books assume that you know how to program in C.

For a complete understanding of OpenGL, you need both types of information. Another difference between the two books is that most of the content of this manual is organized alphabetically, based on the assumption that you know what you don't know and therefore need only to look up a
description of a particular command. The OpenGL Programming Guide is organized like a tutorial: It
explains the simpler OpenGL concepts first and builds up to the more complex ones. Although you
don't have to read the OpenGL Programming Guide to understand the command descriptions in this
manual, your understanding of their intended use will be much more complete if you have read it.

Note that this manual uses abbreviations for command names. Many OpenGL commands are just
variations of each other. For simplicity, only the base name of the command is used, and an asterisk
is included to indicate that there may be more to the actual command name than is being shown.
For example, glVertex* stands for all variations of the command available to specify vertices.

The commands differ mostly in the data type of arguments. Some commands differ in the number
of related arguments and whether those arguments can be specified as a vector or whether they must
be specified separately in a list. For example, if you use the glVertex2f command, you must supply x
and y coordinates as floating-point numbers; with glVertex3sv, you must supply an array of three
short integer values for x, y, and z.

I'm giving this book three stars because even though it's handy to have a reference manual such as
this, I think they could have spent the time to do some cross referencing and some better graphical
structuring to add value to the material. O'Reilly, for example, has some books (particularly the
Nutshell books) which are little more than API references, but people pay for the cross-referencing
and the information design. More care could have been taken on this book.

Portions of this page may be (c) 2006 Muze Inc. Some database content may also be provided by
Baker & Taylor Inc. Copyright 1995-2006 Muze Inc. For personal non-commercial use only. All rights
reserved. Content for books is owned by Baker & Taylor, Inc. or its licensors and is subject to
copyright and all other protections provided by applicable law.

This book provides definitive information on OpenGL, a powerful software interface for graphics
hardware that enables graphics programmers to produce high-quality color images of 3D objects.
The authors' coverage ranges from basic functions of the latest OpenGL release to the sophisticated
capabilities of the OpenGL Utility Library. The third edition has been extensively updated to include
the newest features of OpenGL, Version 1.2, and includes many code examples and sample color
images. The Architecture Review Board (ARB) is an industry consortium responsible for guiding the
evolution of OpenGL and related technologies, and is comprised of industry leaders such as Evans
& Sutherland, Hewlett-Packard, IBM, Intel, Intergraph, Microsoft, Sun Microsystems, and Silicon
Graphics.

This book also includes a conceptual overview of OpenGL, a summary of commands and routines, a
chapter on defined constants and associated commands, and descriptions of the multitexturing and
imaging subset ARB extensions. The OpenGL Technical Library provides tutorial and reference
books for OpenGL.

Provides an overview of the commands and routines in the OpenGL utility library (GLU) and the
OpenGL extensions to the X window system (GLX). Intended as a companion to the , the reference
alphabetically lists 160 commands, identifying each command's parameters, the effects of the
command, and what errors might result from using the command. The fourth edition is updated for
OpenGL versions 1.3 and 1.4.

Now in its fourth edition, this definitive documentation of all core OpenGL functions is completely
revised and updated for OpenGL Versions 1.3 and 1.4. Featuring coverage of cube-mapped
textures, multisampling, depth textures and shadowing, multitexturing, and register combiners, this
book also documents all OpenGL Utility Library functions (GLU 1.3) and the OpenGL extension to
the X Window System (GLX 1.3).

This manual is intended as the companion reference volume to the third edition of the OpenGL
Programming Guide. It provides an introduction to OpenGL ... Show synopsis This manual is
intended as the companion reference volume to the third edition of the OpenGL Programming
Guide. It provides an introduction to OpenGL, its commands and routines, and the constants defined in OpenGL and the commands that use them. Most of the book consists of reference pages, describing each set of commands for OpenGL, GLU, and GLX; their parameters; effects of the command; and what errors might result from using the command. Hide synopsis


Description: Fine. 0201657651 Used, but looks brand new. Only very slight... Fine. 0201657651 Used, but looks brand new. Only very slight signs of use. Cover and binding are undamaged, and pages are crisp and unmarked. Unbeatable customer service, and we usually ship the same or next day. Over one million satisfied customers!

Description: Good. Giving great service since 2004: Buy from the Best! 4, 000... Good. Giving great service since 2004: Buy from the Best! 4, 000, 000 items shipped to delighted customers. We have 1, 000, 000 unique items ready to ship! Find your Great Buy today! Trade paperback (US). Glued binding. 704 p.

Description: Very Good. DON'T WASTE YOUR $$ ON THE OVERPRICED NEW EDITION! !... Very Good. DON'T WASTE YOUR $$ ON THE OVERPRICED NEW EDITION! ! ! ! This recent edition is guaranteed to have the same exact information or you can return it for a full refund. This book is packed and ready to ship today. It is in good or better condition, however all textbooks may contain some highlighting. Since all our books are previously owned, we cannot guarantee the supplemental materials.

Copyright in bibliographic data and cover images is held by Nielsen Book Services Limited, Baker & Taylor, Inc., or by their respective licensors, or by the publishers, or by their respective licensors. For personal use only. All rights reserved. All rights in images of books or other publications are reserved by the original copyright holders.