国外计算机科学教材系列

PEARSON
Prentice Hall

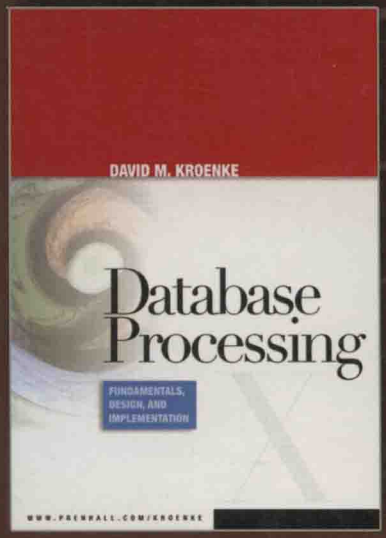
数据库处理

——基础、设计与实现（第十版）

Database Processing

Fundamentals, Design, and Implementation

Tenth Edition



英文版

[美] David M. Kroenke 著

電子工業出版社
Publishing House of Electronics Industry
http://www.phei.com.cn

经典教材

数据库处理

——基础、设计与实现（第十版）

英文版

经典数据库教材！这一版本包括了最新的数据库实用技术和技能

这是本书的第十版，虽然我们对数据库基础理论和技术的讲述已经非常成熟。但作者为了适应教学环境的最新变化，这一版本对全书的结构和内容做了很多重大改变。其中最重要的改变就是突破了传统数据库教材“从数据模型讲述数据库设计”的陈规，强调学习过程中的乐趣，让读者从一开始就能把所学的知识用于解决具体的应用实例。

本书的特点与优势

- 让读者立即体验增强的 SQL（第 2 章）
- 以三种方式创建数据库：
  - 对现有的数据应用规范化
  - 转换 E-R 数据模型
  - 再设计现有的数据库
- 学会“鸦脚”E-R 模型即业界标准的 E-R 模型
- 学会从 Microsoft ASP 和 Java JSP 中访问数据库
- 使用最新的技术，如 XML 和 ADO .NET 等
- 将数据库知识应用于数据挖掘，包括：
  - 市场一揽子分析
  - RFM 分析

本书添加了大量专业的参考文献，并以完整的章节讲述了 SQL Server，Oracle，Microsoft Access 以及 MySQL 的应用。本书必将成为读者职业生涯中的利器！

David M. Kroenke 是一位多产的计算机畅销书作家。他撰写或与人合著的图书专业性强，涉及面广。注重专业理论与实践运用的结合，精通数据库处理、数据库概念、商用计算机系统和管理信息系统等。

ISBN 7-121-02606-6
9 787121 026065 >

传播教育信息 共享教育资源
华信教育资源网
www.hxedu.com.cn
欢迎登录 获取优质教学资源

PEARSON Education

PEARSON Prentice Hall

PHEI

责任编辑：李秦华
责任美编：毛惠庚

ISBN 7-121-02606-6　定价：68.00 元

PEARSON Prentice Hall

国外计算机科学教材系列

数据库处理——基础、设计与实现

Database Processing: Fundamentals, Design, and Implementation, Tenth Edition

（第十版）

英文版

社

国外计算机科学教材系列

数据库处理

—— 基础、设计与实现

（第十版）（英文版）

Database Processing

Fundamentals, Design, and Implementation

Tenth Edition

[美] David M. Kroenke 著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内容简介

本书从基础、设计和实现三个层面介绍数据库处理技术，内容全面详实，既包括数据库设计、数据库实现、多用户数据处理、数据访问标准等经典理论，也包括商务智能、XML 和 .NET 等最新技术。

本书在内容编排和写作风格新颖，强调学习过程中的乐趣，围绕两个贯串全书的项目练习，让读者从一开始就能把所学的知识用于解决具体的应用实例。

本书每章都有丰富的习题，可作为高校本科生或研究生的数据库课程的双语教材，同时也是一本很好的专业参考书。

版权贸易合同登记号　图字：01-2006-2941

图书在版编目（CIP）数据

数据库处理：基础、设计与实现：第 10 版 /（美）克罗恩克（Kroenke, D. M.）著
北京：电子工业出版社，2006.6
（国外计算机科学教材系列）
书名原文：Database Processing: Fundamentals, Design, and Implementation, Tenth Edition
ISBN 7-121-02606-6

Ⅰ. 数...　Ⅱ. 克...　Ⅲ. 数据库 - 教材 - 英文　Ⅳ. TP311.13

中国版本图书馆 CIP 数据核字（2006）第 044706 号

责任编辑：李秦华
印　　刷：北京市顺义兴华印刷厂
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编：100036
经　　销：各地新华书店
开　　本：787 × 980　1 /16　　印张：43.5　　字数：1114 千字
印　　次：2006 年 6 月第 1 次印刷
定　　价：68.00 元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换；若书店售缺，请与本社发行部联系。联系电话：（010）68279077。质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

出 版 说 明

21世纪初的5至10年是我国国民经济和社会发展的重要时期，也是信息产业快速发展的关键时期。在我国加入WTO后的今天，培养一支适应国际化竞争的一流IT人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡，是我国面对国际竞争时成败的关键因素。

当前，正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期，为使我国教育体制与国际化接轨，有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材，以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验，翻译出版了“国外计算机科学教材系列”丛书，这套教材覆盖学科范围广、领域宽、层次多，既有本科专业课程教材，也有研究生课程教材，以适应不同院系、不同专业、不同层次的师生对教材的需求，广大师生可自由选择和自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时，我们也适当引进了一些优秀英文原版教材，本着翻译版本和英文原版并重的原则，对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上，我们大都选择国外著名出版公司出版的高校教材，如Pearson Education培生教育出版集团、麦格劳－希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者，如道格拉斯·科默（Douglas E. Comer）、威廉·斯托林斯（William Stallings）、哈维·戴特尔（Harvey M. Deitel）、尤利斯·布莱克（Uyless Black）等。

为确保教材的选题质量和翻译质量，我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士，也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中，为提高教材质量，我们做了大量细致的工作，包括对所选教材进行全面论证；选择编辑时力求达到专业对口；对排版、印制质量进行严格把关。对于英文教材中出现的错误，我们通过与作者联络和网上下载勘误表等方式，逐一进行了修订。

此外，我们还将与国外著名出版公司合作，提供一些教材的教学支持资料，希望能为授课老师提供帮助。今后，我们将继续加强与各高校教师的密切联系，为广大师生引进更多的国外优秀教材和参考书，为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

PREFACE

While preparing the tenth edition of this text, I decided to make major changes to the text's organization and content. The basic structure of the first nine editions was designed for a teaching environment that no longer exists. Unlike the early years of database processing, today students have ready access to data modeling and DBMS products. Furthermore, today's students are too impatient to start a class with lengthy conceptual discussions on data modeling and database design. They want to do something, see a result, and obtain feedback. Also, in the current economy, students need to reassure themselves that they are learning marketable skills.

SQL

Given these changes in the classroom environment, the first change I made was to move SQL forward, all the way up to Chapter 2. Actually, I moved just the presentation of SQL SELECT statements to Chapter 2, leaving the discussion of SQL DDL and other DML statements for Chapters 7 and 8.

By presenting SQL SELECT statements in Chapter 2, students learn early in the class how to query data and obtain results, seeing firsthand some of the ways that database technology will be useful to them.

The text assumes that students will work through the SQL statements and examples with a DBMS product. This is practical today, because every student has access to Microsoft Access. The text can be purchased with versions of SQL Server and Oracle as well. Alternatively, MySQL is available to students as a free download. Thus, students can actively use a DBMS product by the end of the first week of class.

The presentation and discussion of SQL is spread over three chapters so that students can learn about this important topic in small bites. SQL SELECT statements are taught in Chapter 2. SQL DDL and SQL DML statements are presented in Chapter 7. Correlated subqueries and EXISTS/NOT EXISTS statements are described in Chapter 8. Each topic appears in the context of accomplishing practical tasks. Correlated subqueries, for example, are used to verify functional dependency assumptions, a necessary task for database redesign.

This box illustrates another new feature of this edition: BTW boxes are used to separate comments from the text discussion. Sometimes they present ancillary material; other times they reinforce important concepts.

A Spiral Approach to Database Design

Today, databases arise from three sources: (1) from the integration of existing data from spreadsheets, data files, and database extracts; (2) from the development of new information systems projects, and (3) from the need to redesign an existing database to adapt to changing requirements. As I thought about these three sources, I realized that they present instructors with a significant pedagogical opportunity. Rather than teach database design just once from data models, why not teach database design three times, once for each of these sources? This idea turned out to be even more successful than I expected.

Design Iteration 1: Databases from Existing Data

Considering the design of databases from existing data, I asked myself, if someone were to email me a set of tables and say, "Create a database from them," how would I proceed? I would examine the tables in light of normalization criteria and then determine whether the new database was for query only or whether it was for query and update. Depending on the answer, I would denormalize the data, joining them together, or I would normalize the data, pulling them apart. All of which is important for students to know and understand.

Therefore, the first iteration of database design gives instructors a rich opportunity to teach normalization, not as a set of theoretical concepts, but rather as a useful toolkit for making design decisions for databases created from existing data. Additionally, as I've learned from recent data mining consulting experiences, the construction of databases from existing data is an increasingly common task that is often assigned to junior staff members. Learning how to apply normalization to the design of databases from existing data not only provides an interesting way of teaching normalization, it is also common and useful!

Furthermore, large organizations are increasingly licensing standardized software from vendors such as SAP, Oracle, and Siebel. Such software already has a database design. But with every organization running the same software, many are learning that they can only gain a competitive advantage if they make better use of the data in those predesigned databases. Hence, students who know how to extract data and create read-only databases for reporting and data mining have obtained marketable skills in the world of ERP and other packaged software solutions.

Design Iteration 2: Data Modeling and Database Design

The second source of databases is from new systems development. Although not as common as in the past, many databases are still created from scratch. Thus, students still need to learn data modeling, and they still need to learn how to transform data models into database designs.

Semantic Object Modeling Moved to Appendix

I lost the battle for the semantic object model (SOM). The entity-relationship (E-R) model had too much market momentum for SOM to overcome it. Also, a new data modeling technique requires the support of a major player like Microsoft or Oracle, and we were never able to obtain that support. And, I have to admit the possibility that SOM was not significantly better than the E-R model. Hard for me to believe, but maybe so. In any case, SOM is presented in Appendix E, but the E-R data model is used everywhere else in the text. With that decision, the next was which version of E-R to use.

IDEF1X Has Been Replaced by the Crow's Foot Model

The ninth edition of this text used IDEF1X extensively, but since then, I've concluded that IDEF1X is not worth the trouble it causes. The essence of E-R modeling is more easily taught using the crow's foot version of the E-R model, and I have used it throughout.

IDEF1X is explained, however, in Appendix B in case your students will graduate into an environment where it is used. At one point, I thought IDEF1X might be worth teaching because it is a national standard, but Jack Becker at the University of North Texas put the nail in the coffin when he said, "So was Ada."

This text teaches and uses the simple, plain vanilla, crow's foot version of the E-R model.

BTW

In my opinion, the best E-R data modeling tool for teaching is ERwin from Computer Associates. A free version of ERwin with a 60-day license can be obtained from *www.ca.com*. Search for the "All Fusion ERwin download."

I prefer ERwin over Visio for two reasons. One, Visio is a general-purpose graphics program that has a data modeling template. Accordingly, most of the error messages and help text is generalized, confusing, and unhelpful. ERwin is built specifically for data modeling. Unlike Visio, you cannot use it to design your kitchen or garden, but you can build comprehensive data models with it.

Second, Visio is more of a table modeling than a data modeling tool. In particular, it is not possible to represent an N:M strong entity relationship in Visio. Instead, the intersection table must be constructed and modeled. This confounds data modeling with database design in just the way that I am attempting to teach my students to avoid.

This does not mean that you must use ERwin to use this book. You can use Visio or any other E-R data modeling product. But, if you're looking for a tool to use, I recommend ERwin.

I have no association, in any way, with Computer Associates. I do not own their stock, I do not consult for their engineers, I do not teach their seminars. I just like their product.

Database Design from E-R Data Models

As discussed in Chapter 6, designing a database from data models consists of three tasks: representing entities and attributes with tables and columns; representing maximum cardinality by creating and placing foreign keys; and representing minimum cardinality via constraints, triggers, and application logic.

The first two tasks are straightforward. However, designs for minimum cardinality are more difficult. Required parents are easily enforced using NOT NULL foreign keys and referential integrity constraints. Required children are more problematic. I have simplified the discussion of this topic from the ninth edition, however, by limiting the use of referential integrity actions and by supplementing those actions with design documentation. See the discussion around Figure 6-27.

Although the design for required children is complicated, it is important for students to learn. It also provides a reason for students to learn about triggers as well. In any case, the discussion of these topics is much simpler than it was in prior editions because of the use of the crow's foot model and the use of ancillary design documentation.

Design Iteration 3: Database Redesign

Database redesign is both common and difficult. As stated in Chapter 8, information systems cause organizational change. New information systems give users new behaviors, and as users behave in new ways, they require changes in their information systems.

Database redesign, the third iteration of database design, is by nature complex. Depending on your students, you may wish to skip it, and you can do so without loss of continuity.

Database redesign is presented after the discussion of SQL DDL and DML in Chapter 7 because it requires the use of advanced SQL. It also provides a practical reason to teach correlated subqueries and EXISTS/NOT EXISTS statements.

Business Intelligence: Reporting and Data Mining

Another new feature of this text is a new chapter (Chapter 15) on business intelligence (BI) systems. The chapter includes a discussion on data management for data warehouses and data marts. It also describes reporting and data data mining applications, including OLAP.

Chapter 15 presents two applications that should be particularly interesting to students. The first is RFM analysis, a reporting application frequently used by mail order and e-commerce companies. The complete RFM analysis is accomplished in Chapter 15 through the use of standard SQL-92 statements. Additionally, this chapter includes a market-basket analysis that is also performed using SQL correlated subqueries. This chapter can be assigned at any point after Chapter 8 and could be used as a motivator to illustrate the practical applications of SQL midcourse.

Active Use of a DBMS Product

As stated earlier, this edition assumes that the students will actively use a DBMS product. The question is, which one? Realistically, most of us have four alternatives to consider: Microsoft Access, Oracle, Microsoft SQL Server, or MySQL. You can use any of those products with this text, and tutorials for each of them are presented in Appendix A and Chapters 10, 11, and 14, respectively.

Given the limitations of class time, I have found it necessary to pick and use just one of these products. I usually devote a portion of a lecture to discussing the characteristics of each, but I have found it best to limit student work to one of them.

Using Access

The primary advantage of Access is accessibility. Most students already have a copy, and if not, copies are easily obtained. Many students will have used Access in their introductory or other classes. Appendix A is a tutorial on Access for students who have not used it but who wish to use it with this book.

However, Access has several disadvantages. First, as explained in Chapter 1, Access is a combination application generator and DBMS. Access confuses students because it confounds database processing with application development. Also, Access hides SQL behind its query processor and makes SQL appear as an afterthought rather than a foundation. Furthermore, as discussed in Chapter 2, Access does not correctly process some of the basic, SQL-92 standard statements. Finally, Access does not support triggers. You can simulate triggers by trapping Windows events, but that technique is nonstandard and it miscommunicates the nature of trigger processing.

Using Oracle, SQL Server, or MySQL

Which of these products to use depends on your local situation. Oracle, a superb enterprise-class DBMS product, is difficult to install. However, if you have local staff to support your students, it can be an excellent choice. Oracle's SQL*Plus is a handy tool for learning SQL, triggers, and stored procedures, as shown in Chapter 10. In my experience,

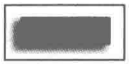
students require considerable support to install Oracle on their own computers, and you may be better off to use Oracle from a central server.

SQL Server, although probably not as robust as Oracle, is easy to install on Windows machines, and it provides the capabilities of an enterprise-class DBMS product. It can be driven from Visual Studio .NET, but I use the Enterprise Manager and Query Analyzer. SQL Server can be used to learn SQL, triggers, and stored procedures, as shown in Chapter 11.

By the time you read this, Microsoft may have released SQL Server 2005, and it promises to have many new and important features. It also appears to be far more complicated than the current version of SQL Server, so be prepared to make adjustments. T-SQL is still supported and the knowledge students learn with SQL Server 2000 will be applicable to the 2005 version.

MySQL is an open-source DBMS product that is receiving increased attention and market share. The capabilities of MySQL are continually being upgraded, and you can now write stored procedures. However, MySQL does not support triggers, which I view as a significant limitation. Still, it's an excellent product and fun to use. See Chapter 14 for its use with Java Server Pages.

If the DBMS you use is not driven by local circumstances and you do have a choice, I recommend using SQL Server. It has all of the features of an enterprise-class DBMS product, and it is easy to install and use. You can order this text with a shrink-wrapped version of SQL Server and a 90-day license. You also can order the text with a shrink-wrapped version of Oracle.



Overview of the Chapters in the Tenth Edition

Chapter 1 sets the stage by introducing database processing, describing basic components of database systems, and summarizing the history of database processing. Chapter 2 presents SQL SELECT statements. It also includes sections on how to submit SQL statements to Access, Oracle, and SQL Server. If the students are using Access for the first time, they will also need to study Appendix A at this point.

The next four chapters, Chapters 3 through 6, present the first two iterations of database design. Chapter 3 presents the principles of normalization using Boyce-Codd normal form. It describes the problems of multivalued dependencies and explains how to eliminate them. This foundation in normalization is applied in Chapter 4 to the design of databases from existing data.

Chapters 5 and 6 describe the design of new databases. Chapter 5 presents the E-R data model. Traditional E-R symbols are explained, but the majority of the chapter uses the crow's foot notation. Chapter 5 provides a taxonomy of entity types, including strong, ID-dependent, weak but not ID-dependent, subtype, and recursive. The chapter concludes with a simple modeling example for a university database.

Chapter 6 describes the transformation of data models into database designs by converting entities and attributes to tables and columns, by representing maximum cardinality by creating and placing foreign keys, and by representing minimum cardinality via DBMS constraints, triggers, and application code. The primary section of this chapter parallels the entity taxonomy in Chapter 5.

Chapter 7 presents SQL DDL and DML. SQL DDL is used to implement the design of an example introduced in Chapter 6. INSERT, UPDATE, and DELETE statements are discussed, as are SQL views. Additionally, the principles of embedding SQL in program code are presented, and triggers and stored procedures are explained.

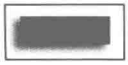
Database redesign, the third iteration of database design, is described in Chapter 8. This chapter presents SQL correlated subqueries and EXISTS/NOT EXISTS statements and uses those statements in the redesign process. Reverse engineering is described, and basic redesign patterns are illustrated and discussed.

Chapters 9, 10, and 11 consider the management of multiuser organizational databases. Chapter 9 describes database administration tasks, including concurrency, security, and backup and recovery. Chapters 10 and 11 then describe Oracle and SQL Server, respectively. These chapters show how to use these products to create database structures and process SQL statements. They also explain concurrency, security, and backup and recovery with each product. The discussion in Chapters 10 and 11 parallels the order of discussion in Chapter 9.

Chapters 12, 13, and 14 address standards for accessing databases. Chapter 12 presents ODBC, OLE DB, ADO, and ASP. It illustrates the use of these technologies for the publication of both SQL Server and Oracle databases via Active Server Pages. Chapter 13 describes the integration of XML and database technology. The chapter begins with a primer on XML and then shows how to use the FOR XML SQL statement in SQL Server. The chapter concludes with a Visual Basic .NET application that uses ADO.NET to create a dataset from the tables in an Oracle database.

Chapter 14 presents data access standards for the open source/Java world. It describes JDBC and discusses and illustrates Java Server Pages. It concludes with a survey of MySQL.

Chapter 15 concludes the text with a discussion of business intelligence systems, data warehouses, and data marts. It illustrates the use of SQL for RFM reporting analysis and for market-basket analysis.



Supplements: *www.prenhall.com/kroenke*

This text is accompanied by a wide variety of supplements. Please visit the text's Web site at ***www.prenhall.com/kroenke*** to access the instructor and student supplements described below. The Instructor Resources are also available on CD-ROM. Please see your Prentice Hall representative for more details. All supplements were written by David Auer of Western Washington University.

For Students

- An *Interactive Study Guide* with multiple-choice, true/false, and essay questions. Students receive automatic feedback to their answers. Responses to the essay questions and results from the multiple-choice and true/false questions can be emailed to the instructor after a student finishes a quiz.
- *PowerPoint Presentation Slides* highlight key terms and concepts.
- All of the *Sample Databases* used in this text are available in Access, SQL Server, and Oracle format.
- *Glossary*

For Instructors

- The *Instructor's Resource Manual* provides sample course syllabi, teaching suggestions, and answers to end-of-chapter review, project, and case questions.
- The *Test Item File* and *TestGen* include an extensive set of test questions in multiple-choice, true/false, fill-in-the-blank, short-answer, and essay format. The difficulty level and where the topic is covered in the text are noted for each question. The Test Item File is available in Microsoft Word and in TestGen. TestGen is a comprehensive suite of tools for testing and assessment. It enables instructors to easily create and distribute tests for their courses, either by printing and distributing them

through traditional methods or by online delivery via a LAN. TestGen features Screen Wizards to assist you as you move through the program, and the software is backed with full technical support.

- *PowerPoint Presentation Slides* feature lecture notes that highlight key terms and concepts. Instructors can customize the presentation by adding their own slides or editing the existing ones.
- The *Image Library* is a collection of the text art organized by chapter. This includes all figures, tables, and screenshots (as permission allows) to enhance class lectures and PowerPoint presentations.

Materials for Your Online Course

Prentice Hall supports our adopters using online courses by providing files ready for upload into both WebCT and BlackBoard course management systems for our testing, quizzing, and other supplements. Please contact your local PH representative or ***mis_service@prenhall.com*** for further information on your particular course.

Acknowledgments

I am grateful for the support of many people in the development of this tenth edition. David Auer at Western Washington University has provided invaluable assistance, not only in creating the Instructor's Manual, Test Item File, and PowerPoint slides, but also by helping me shape the organization of this edition and by providing a detailed and insightful review of the manuscript. I would also like to thank Rick Mathieu at James Madison University for interesting and insightful discussions on the database course.

Professor Doug MacLachlan from the Marketing Department at the University of Washington was most helpful to me in understanding the goals, objectives, and technology of data mining, particularly as it pertains to marketing. I would also like to thank Chris Wilkins for his help in testing my Java code. Don Nilson of the Microsoft Corporation helped me understand the importance of XML to database processing.

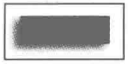
In addition, I wish to thank the reviewers of this edition:

Richard Chrisman, *Northeast Community College*
Richard Clemens, *West Virginia Wesleyan College*
Thomas A. Easton, *Thomas College*
Robert Hofkin, *Goldey-Beacom College*
Douglas M. Kline, *University of North Carolina at Wilmington*
James Marlatt, *University of Colorado at Boulder*
Michael L. Monroe, *Virginia Western Community College*
Tina Ostrander, *Highline Community College*
Daria Santerre, *Norwalk Community College*
Jack Warner, *University of Oklahoma*
Daniel Zhu, *Iowa State University*

Finally, I wish to thank my editor, Bob Horan, for his advice and assistance through three editions, so far(!), of this text. I also thank Lori Cerreto for keeping this project on track. Most importantly, I wish to thank my wife Lynda for her love and support through so many of these writing projects!

David Kroenke
Seattle, Washington

ABOUT THE AUTHOR

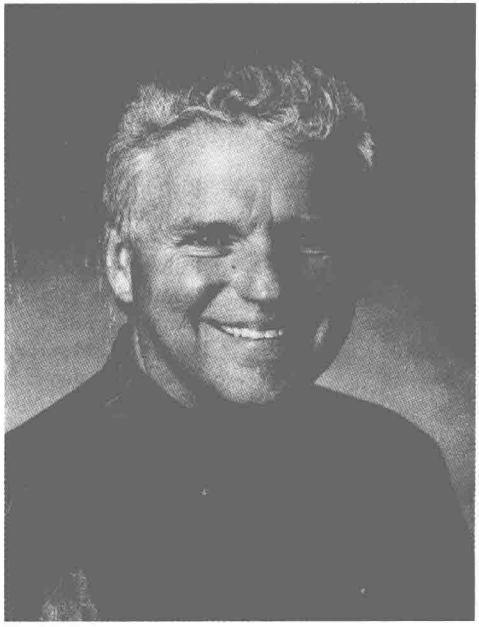


David M. Kroenke

Work Experience

David M. Kroenke has more than 35 years experience in the computer industry. He began as a computer programmer for the U.S. Air Force, working both in Los Angeles and at the Pentagon, where he developed one of the world's first DBMS products while part of a team that created a computer simulation of World War III. That simulation served a key role for strategic weapons studies during a 10-year period of the Cold War.

From 1973 to 1978, Kroenke taught in the College of Business at Colorado State University. In 1977, he published the first edition of *Database Processing,* a text that is currently published in its tenth edition. In 1978, he left Colorado State and joined Boeing Computer Services, where he managed the team that designed database management components of the IPAD project. After that, he joined with Steve Mitchell to form Mitchell Publishing and worked as an editor and author, developing texts, videos, and other educational products and seminars. Mitchell Publishing was acquired by Random House in 1986. During these years he also worked as an independent consultant, primarily as a database disaster repairman helping companies recover from failed database projects.



In 1982, Kroenke was one of the founding directors of the Microrim Corporation. From 1984 to 1987, he served as the Vice President of Product Marketing and Development and managed the team that created and marketed the DBMS product R:base 5000 as well as other related products.

For the next five years, Kroenke worked independently while he developed a new data modeling language called the *semantic object model.* He licensed this technology to the Wall Data Corporation in 1992 and then served as the Chief Technologist for Wall Data's SALSA line of products. He was awarded three software patents on this technology.

Since 1998, Kroenke has continued consulting and writing. His current interests concern the practical applications of data mining techniques

on large organizational databases. An avid sailor, he wrote *Know Your Boat: The Guide to Everything That Makes Your Boat Work,* which was published by McGraw-Hill in 2002.

Consulting

Kroenke has consulted with numerous organizations during his career. In 1978, he worked for Fred Brooks, consulting with IBM on a project that became the DBMS product DB2. In 1989, he consulted for the Microsoft Corporation on a project that became Microsoft Access. In the 1990s, he worked with Computer Sciences Corporation and with General Research Corporation for the development of technology and products that were used to model all of the U.S. Army's logistical data as part of the CALS project. Additionally, he has consulted for Boeing Computer Services, the United States Air Force Academy, Logicon Corporation, and other smaller organizations.

Publications

- *Database Processing,* Prentice Hall, ten editions, 1977–present
- *Database Concepts,* Prentice Hall, two editions, 2004
- *Know Your Boat: The Guide to Everything that Makes Your Boat Work,* McGraw-Hill, 2002
- *Management Information Systems,* Mitchell Publishing/Random House, three editions, 1987–1992
- *Business Computer Systems,* Mitchell Publishing/Random House, five editions, 1981–1990
- *Managing Information for Microcomputers,* co-author with Donald Nilson, Microrim Corporation, 1984
- *Database Processing for Microcomputers,* co-author with Donald Nilson, Science Research Associates, 1985
- *Database: A Professional's Primer,* Science Research Associates, 1978

Teaching

Kroenke taught in the College of Business at Colorado State University from 1973 to 1978. He also has taught part-time in the Software Engineering program at Seattle University. From 1990 to 1991, he served as the Hanson Professor of Management Science at the University of Washington. He currently teaches at the University of Washington. During his career, he has been a frequent speaker at conferences and seminars for computer educators. In 1991, the International Association of Information Systems named him Computer Educator of the Year.

Education

B.S., Economics, United States Air Force Academy, 1968
M.S., Quantitative Business Analysis, University of Southern California, 1971
PhD, Engineering, Colorado State University, 1977

Personal

Kroenke is married, lives in Seattle, and has two grown children and two grandchildren. He enjoys skiing, sailing, and building small boats. His wife tells him he enjoys gardening as well.