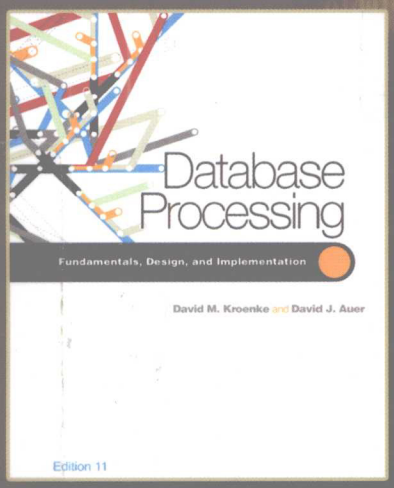
国外计算机科学教材系列

PEARSON

数据库处理
——基础、设计与实现（第十一版）

Database Processing
Fundamentals, Design, and Implementation
Edition 11



英文版

[美] David M. Kroenke
David J. Auer 著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

国外计算机科学教材系列

数据库处理

——基础、设计与实现

（第十一版）

（英文版）

Database Processing

Fundamentals, Design, and Implementation

Edition 11

[美] David M. Kroenke
David J. Auer 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，内容全面翔实，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上新颖，强调学习过程中的乐趣，围绕两个贯穿全书的项目练习，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：01-2010-1389

图书在版编目（CIP）数据

数据库处理：基础、设计与实现：第 11 版：英文/（美）克罗恩克（Kroenke, D. M.），（美）奥尔（Auer, D. J.）著. —北京：电子工业出版社，2010.7
（国外计算机科学教材系列）
ISBN 978-7-121-11241-6

I. ①数… II. ①克…②奥… III. ①数据库－高等学校－教材－英文 IV. ①TP311.13

中国版本图书馆 CIP 数据核字（2010）第 125530 号

策划编辑：谭海平
责任编辑：史　平
印　　刷：北京市天竺颖华印刷厂
装　　订：三河市鑫金马印装有限公司
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
开　　本：787×1092　1/16　印张：39.75　字数：1017 千字
印　　次：2010 年 7 月第 1 次印刷
定　　价：75.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。
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。
服务热线：（010）88258888。

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

Fundamentals, Design, and Implementation

The 11th edition of *Database Processing* refines the organization and content of this classic textbook to reflect a new teaching and professional workplace environment. The organization and topic selection of the 11th edition is designed to:

- Present an early introduction to SQL queries.
- Use a "spiral approach" to database design.
- Use a consistent, generic Information Engineering (IE) Crow's Foot E-R diagram notation for data modeling and database design.
- Provide a detailed discussion of specific normal forms within a discussion of normalization that focuses on pragmatic normalization techniques.
- Use current DBMS technology: Microsoft Access 2007, Microsoft SQL Server 2008, Oracle Database 11g, and MySQL 5.1.
- Create Web database applications based on widely used Web development technology.
- Provide an introduction to business intelligence (BI) systems.
- Discuss the dimensional database concepts used in database designs for data warehouses and OnLine Analytical Processing (OLAP).

These changes have been made because it has become obvious that the basic structure of the earlier editions (up to and including the 9th edition—the 10th edition introduced many of the changes we use in the 11th edition) was designed for a teaching environment that no longer existed. The structural changes to the book were made for several reasons:

- Unlike the early years of database processing, today's students have ready access to data modeling and DBMS products.
- 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.
- In the current economy, students need to reassure themselves that they are learning marketable skills.

Early Introduction of SQL DML

Given these changes in the classroom environment, this book provides an early introduction to SQL data manipulation language (DML) SELECT statements. The discussion of SQL data definition language (DDL) and additional DML statements occurs in 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 nearly every student has access to Microsoft Access. Therefore, Chapters 1 and 2 and Appendix A are written to support an early introduction of Microsoft Access 2007 and the use of Access 2007 for SQL queries (Access 2007 QBE query techniques are also covered).

If a non–Access-based approach is desired, versions of SQL Server 2008, Oracle Database, and MySQL 5.1 are readily available for use. Free versions of the three major DBMS products covered in this book (SQL Server 2008 Express, Oracle Express 10g, and MySQL 5.1 Community Edition) are available for download. Further, the text can be purchased with a licensed educational version of Oracle Database 11g personal edition (this is a developer license) as well. Alternatively, a trial copy of MySQL 5.1 Enterprise Edition is also available as a download. Thus, students can actively use a DBMS product by the end of the first week of class.

BY THE WAY 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 feature used in this book: 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. We believe that the fact that these three sources exist 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? In practice, this idea has turned out to be even more successful than expected.

Design Iteration 1: Databases from Existing Data

Considering the design of databases from existing data, if someone were to e-mail us a set of tables and say, "Create a database from them," how would we proceed? We 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, we would denormalize the data, joining them together, or we 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, 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!

We prefer to teach and use a pragmatic approach to normalization, and present this approach in Chapter 3. However, we are aware that many instructors like to teach normalization in the context of a step-by-step normal form presentation (1NF, 2NF, 3NF, then BCNF), and Chapter 3 includes material to support this approach as well.

In today's workplace, 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.

The IE Crow's Foot Model as a Design Standard

This edition uses a generic, standard IE Crow's Foot notation. Your students should have no trouble understanding the symbols and using the data modeling or database design tool of your choice.

IDEF1X (which was used as the preferred E-R diagram notation in the 9th edition of this text) is explained in Appendix B in case your students will graduate into an environment where it is used, or if you prefer to use it in your classes. UML is explained in Appendix C in case you prefer to use UML in your classes.

BY THE WAY The choice of a data modeling tool is somewhat problematic. The two most readily available tools, Microsoft Visio and Sun Microsystems MySQL Workbench, are database design tools, not data modeling tools. Neither can produce an N:M relationship as such (as a data model requires), but have to immediately break it into two 1:N relationships (as database design does). Therefore, the intersection table must be constructed and modeled. This confounds data modeling with database design in just the way that we are attempting to teach students to avoid.

To be fair to Visio, it is true that data models with N:M relationships can be drawn using either the standard Visio drawing tools or the Entity Relationship shapes dynamic connector. These are shown in Figure FM-01, where the N:M relationships ENTITY-ONE-to-ENTITY-THREE and ENTITY-FOUR-to-ENTITY-FIVE were drawn using the dynamic connector and show the correct line end symbols.

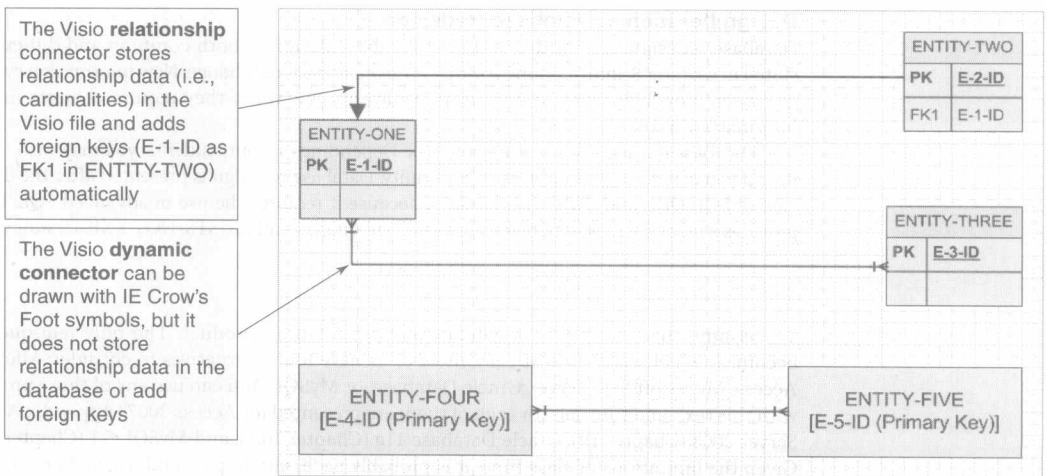
ENTITY-ONE and ENTITY-THREE are Visio database entity objects, and Visio stores object data such as column type, primary key, and so on, about the entities, but *not* about the relationship between them. ENTITY-FOUR and ENTITY-FIVE were drawn using the Visio rectangle drawing tool, and no database object data is stored by these shapes or the relationship between them.

Visio is at its best using the Visio relationship connector to connect entity objects. In this case, as shown in the ENTITY-ONE-to-ENTITY-TWO relationship, relationship data such as minimum and maximum cardinalities is stored in database objects and foreign keys (such as E-1-ID as FK1 in ENTITY-TWO) are automatically inserted in child

(*continued*)

Figure FM-01

Data Models in Microsoft Visio 2007



entities. However, there is no *N:M* relationship connector in Visio, so we cannot model N:M relationships using a relationship connector. Without the ability to store relationship data for N:M relationships, it is problematic to use Visio as data modeling tool while it functions well as database design tool.

There are good data modeling tools available, but they tend to be more complex and expensive. Two examples are Visible Systems' Visible Analyst and Computer Associates' ERwin Data Modeler. Visible Analyst is available in a student edition (at a modest price), and a one-year time-limitedCA ERwin Data Modeler Community Edition suitable for class use can be downloaded from www.ca.com/us/software-trials.aspx—CA has limited the number of objects that can be created by this edition to 25 entities per model, and disabled some other features (see www.ca.com/files/TechnicalDocuments/erwin-community-edition-matrix_197107.pdf), but there is still enough functionality to make this product a possible choice for class use.

Database Design from E-R Data Models

As we discuss 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. In this book, however, we simplify the discussion of this topic by limiting the use of referential integrity actions and by supplementing those actions with design documentation. See the discussion around Figure 6-28.

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 IE Crow's Foot model and the use of ancillary design documentation.

David Kroenke is the creator of the semantic object model (SOM). The SOM is presented in Appendix E; the E-R data model is used everywhere else in the text.

Design Iteration 3: Database Redesign

Database redesign, the third iteration of database design, 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 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.

Active Use of a DBMS Product

We assume that the students will actively use a DBMS product. The only real question becomes "which one?" Realistically, most of us have four alternatives to consider: Microsoft Access, Microsoft SQL Server, Oracle Database, or MySQL. You can use any of those products with this text, and tutorials for each of them are presented for Access 2007 (Appendix A), SQL Server 2008 (Chapter 10), Oracle Database 11g (Chapter 10A), and MySQL 5.1 (Chapter 10B). Given the limitations of class time, it is probably necessary to pick and use just one of these

products. You can often devote a portion of a lecture to discussing the characteristics of each, but it is usually best to limit student work to one of them. The possible exception to this is starting the course with Microsoft Access, and then switching to a more robust DBMS product later in the course.

Using Microsoft Access 2007

The primary advantage of Microsoft 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 2007 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 in its default setup. Finally, Access does not support triggers. You can simulate triggers by trapping Windows events, but that technique is nonstandard and does not effectively communicate the nature of trigger processing.

Using Oracle, SQL Server, or MySQL

Choosing which of these products to use depends on your local situation. Oracle Database 11g, a superb enterprise-class DBMS product, is difficult to install and administer. However, if you have local staff to support your students, it can be an excellent choice. As shown in Chapter 10A, Oracle's SQL Developer GUI tool (or SQL*Plus if you are dedicated to this beloved command-line tool) is a handy tool for learning SQL, triggers, and stored procedures. In our 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 2008, 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. The standard database administrator tool is the Microsoft SQL Server Management Studio GUI tool. As shown in Chapter 10, SQL Server 2008 can be used to learn SQL, triggers, and stored procedures.

MySQL 5.1, discussed in Chapter 10B is an open-source DBMS product that is receiving increased attention and market share. The capabilities of MySQL are continually being upgraded, and MySQL 5.1 supports stored procedures and triggers. MySQL also has excellent GUI tools (MySQL Query Browser and MySQL Administrator) and an excellent command-line tool (the MySQL Command Line Client). It is the easiest of the three products for students to install on their own computers. It also works with the Linux operating system, and is popular as part of the AMP (Apache-MySQL-PHP) package (known as WAMP on Windows and LAMP on Linux).

BY THE WAY If the DBMS you use is not driven by local circumstances and you do have a choice, we recommend using SQL Server 2008. It has all of the features of an enterprise-class DBMS product, and it is easy to install and use. Another option is to start with Access 2007 if it is available, and switch to SQL Server 2008 at Chapter 7. Chapters 1 and 2 and Appendix A are written specifically to support this approach. A variant is to use Access 2007 as the development tool for forms and reports running against an SQL Server 2008 database.

If you prefer a different DBMS product, you can still start with Access 2007 and switch later in the course. You can order the text with a shrink-wrapped version of Oracle, while a trial version of SQL Server can be downloaded from Microsoft, and MySQL is so easy to download (and updated often enough) that it makes no sense to package a copy with the book.

Focus on Database Application Processing

In this edition, we clearly draw the line between *application development* per se and *database application processing*. Specifically, we have:

- Focused on specific database dependent applications:
  - Web-based, database-driven applications
  - XML-based data processing
  - Business intelligence (BI) systems applications
- Emphasized the use of commonly available, multiple OS compatible application development languages.
- Limited the use of specialized vendor-specific tools and programming languages as much as possible.

There is simply not enough room in this book to provide even a basic introduction to, for example, Visual Basic .NET and Java. Therefore, rather than attempting to introduce these languages, we leave them for other classes where they can be covered at an appropriate depth. Instead, we focus on basic tools that are relatively straightforward to learn and immediately applicable to database-driven applications. We use PHP as our Web development language, and we use the readily available Eclipse integrate development environment (IDE) as our development tool. The result is a very focused final section of the book, where we deal specifically with the interface between databases and the applications that use them.

Although we try to use widely available software as much as possible, there are, of course, exceptions where we must use vendor specific tools. For BI applications, for example, we draw on Microsoft Excel's PivotTable capabilities and the Microsoft SQL Server 2008 Data Mining Add-ins for Microsoft Office 2007. However, there are either alternatives to these tools (OpenOffice.org DataPilot capabilities, the Palo OLAP Server) or the tools are generally available for download.

Business Intelligence Systems and Dimensional Databases

This edition increases coverage of business intelligence (BI) systems (Chapter 13). The chapter now includes a discussion of dimensional databases, which are the underlying structure for data warehouses, data marts, and OLAP servers. It still covers data management for data warehouses and data marts and also describes reporting and data mining applications, including OLAP.

Chapter 13 includes coverage of 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 13 through the use of standard SQL 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.

Overview of the Chapters in the 11th Edition

Chapter 1 sets the stage by introducing database processing, describing basic components of database systems, and summarizing the history of database processing. If the students are using Access 2007 for the first time (or need a good review), they will also need to study Appendix A at this point. Chapter 2 presents SQL SELECT statements. It also includes sections on how to submit SQL statements to Access, SQL Server, Oracle, and MySQL.

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 IE Crow's Foot notation. Chapter 5 provides a taxonomy of entity types, including strong, ID-dependent, weak but not ID-dependent, supertype/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 carefully designed 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, 10A, and 10B consider the management of multiuser organizational databases. Chapter 9 describes database administration tasks, including concurrency, security, and backup and recovery. Chapters 10, 10A, and 10B then describe SQL Server 2008, Oracle Database 11g, and MySQL 5.1, 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, 10A, and 10B parallels the order of discussion in Chapter 9 as much as possible, although rearrangements of some topics are made as needed to support the discussion of a specific DBMS product.

We have significantly extended our coverage of Access, SQL Server, Oracle Database, and MySQL in this book. In order to keep the bound book to a reasonable length, and to keep the cost of the book down, we have chosen to provide some materials by download from our Web site at www.pearsonhighered.com/kroenke. There you will find:

- Chapter 10A—Managing Databases with Oracle Database 11g
- Chapter 10B—Managing Databases with MySQL 5.1
- Appendix A—Getting Started with Microsoft Access 2007
- Appendix B—The IDEF1X Standard
- Appendix C—UML-Style Entity-Relationship Diagrams
- Appendix D—Data Structures for Database Processing
- Appendix E—The Semantic Object Model

Chapters 11, 12, and 13 address standards for accessing databases. Chapter 11 presents ODBC, OLE DB, ADO.NET, ASP.NET, JDBC, and Java Server Pages (JSP). It then introduces PHP (and the Eclipse IDE) and illustrates the use of PHP for the publication of databases via Web pages. Chapter 12 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.

Chapter 13 concludes the text with a discussion of BI systems, dimensional data models, data warehouses, and data marts. It illustrates the use of SQL for RFM reporting analysis and for market basket analysis.

Supplements

This text is accompanied by a wide variety of supplements. Please visit the text's Web site at www.pearsonhighered.com/kroenke to access the instructor and student supplements described below. Please contact your Pearson sales representative for more details. All supplements were written by David Auer.

For Students

- Many of the *Sample Databases* used in this text are available in Access, Oracle, SQL Server, and MySQL 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. TestGen conversions for WebCT and BlackBoard course management systems are available.
- *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 Pearson sales representative for further information on your particular course.

Acknowledgments

We are grateful for the support of many people in the development of this 11th edition and previous editions. Thanks to 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 in understanding the goals, objectives, and technology of data mining, particularly as it pertains to marketing. Don Nilson of the Microsoft Corporation helped us understand the importance of XML to database processing.

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

Ann Aksut, *Central Piedmont Community College*

Allen Badgett, *Oklahoma City University*

Rich Beck, *Washington University*

Jeffrey J. Blessing, *Milwaukee School of Engineering*
Jason Deane, *Virginia Polytechnic Institute and State University*
Barry Flaschbart, *Missouri University of Science and Technology*
Dianne Hall, *Auburn University*
Jeff Hassett, *University of Utah*
Barbara Hewitt, *Texas A&M, Kingsville*
William Hochstettler, *Franklin University*
Nitin Kale, *University of Southern California, Los Angeles*
Johnny Li, *South University*
Mike Morris, *Southeastern Oklahoma State University*
Catherine Ricardo, *Iona College*
Kevin Roberts, *DeVry University*
Ioulia Rytikova, *George Mason University*
Christelle Scharff, *Pace University*
K. David Smith, *Cameron University*
Marcia Williams, *Bellevue Community College*

David Kroenke would like to thank our editor, Bob Horan, for his advice and assistance through four editions, so far(!), of this text. I also thank Kelly Loftus 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!

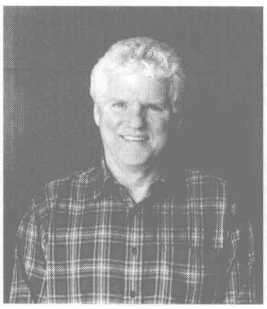
David Auer would like to thank David Kroenke for his generosity, insight, and support during this project. Our editor, Bob Horan; our assistant editor, Kelly Loftus; and our production project manager, Kelly Warsak, for their professionalism, insight, support, and assistance in the development of this project. My wife, Donna, for her love, encouragement, and patience while I was completing this project.

David Kroenke
Seattle, Washington

David Auer
Bellingham, Washington

About the Authors

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 significant and successful textbook that, over 30 years later, you now are reading in its 11th 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 U.S. Air Force Academy, Logicon Corporation, and other smaller organizations.

Publications

- *Database Processing*, Pearson Prentice Hall, 11 editions, 1977–present (coauthor with David Auer, 11th edition)
- *Database Concepts*, Pearson Prentice Hall, four editions, 2004–present (coauthor with David Auer, third and fourth editions)

- *Using MIS*, Pearson Prentice Hall, two editions, 2006–present
- *Experiencing MIS*, Pearson Prentice Hall, two editions, 2007–present
- *MIS Essentials*, Pearson Prentice Hall, one edition, 2009
- *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*, Microrim Corporation, 1984 (coauthor with Donald Nilson)
- *Database Processing for Microcomputers*, Science Research Associates, 1985 (coauthor with Donald Nilson)
- *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. Most recently, he taught at the University of Washington from 2002 to 2008. 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.

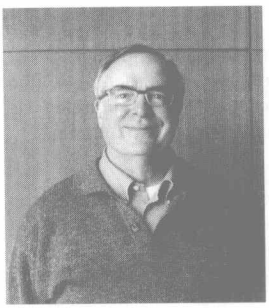
Education

B.S., Economics, U.S. Air Force Academy, 1968
M.S., Quantitative Business Analysis, University of Southern California, 1971
Ph.D., Engineering, Colorado State University, 1977

Personal

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

David J. Auer



Work Experience

David J. Auer has more than 25 years experience teaching college-level business and information systems courses and for the last 15 years has worked professionally in the field of information technology. He served as a commissioned officer in the U.S. Air Force, with assignments to NORAD and the Alaskan Air Command in air defense operations. He later taught both business administration and music classes at Whatcom Community College, and business courses for the Chapman College Residence Education Center at Whidbey Island Naval Air Station. He was a founder of the Puget Sound Guitar Workshop (now in its 35th year of operations). He worked as a psychotherapist and organizational development consultant for the Whatcom Counseling and Psychiatric Clinic's Employee Assistance Program, and provided training for the Washington State Department of Social and Health Services. He has taught for Western Washington University's College of Business and Economics since 1981 and has been the college's Director of Information Systems and Technology Services since 1994.

Publications

- *Database Processing*, Pearson Prentice Hall, one edition, 2009 (coauthor with David Kroenke)
- *Database Concepts*, Pearson Prentice Hall, two editions, 2007–present (coauthor with David Kroenke)

- *Network Administrator: NetWare 4.1*, Course Technology, 1997 (coauthors Ted Simpson and Mark Ciampa)
- *New Perspectives on Corel Quattro Pro 7.0 for Windows 95*, Course Technology, 1997 (coauthors June Jamrich Parsons, Dan Oja, and John Leschke)
- *New Perspectives on Microsoft Excel 7 for Windows 95—Comprehensive*, Course Technology, 1996 (coauthors June Jamrich Parsons and Dan Oja)
- *New Perspectives on Microsoft Office Professional for Windows 95—Intermediate*, Course Technology, 1996 (coauthors June Jamrich Parsons, Dan Oja, Beverly Zimmerman, Scott Zimmerman, and Joseph Adamski)
- *The Student's Companion for Use with Practical Business Statistics* (3rd ed.), Irwin, 1993
- *Microsoft Excel 5 for Windows—New Perspectives Comprehensive*, Course Technology, 1995 (coauthors June Jamrich Parsons and Dan Oja)
- *Introductory Quattro Pro 6.0 for Windows*, Course Technology, 1995 (coauthors June Jamrich Parsons and Dan Oja)
- *Introductory Quattro Pro 5.0 for Windows*, Course Technology, 1994 (coauthors June Jamrich Parsons and Dan Oja)
- *The Student's Companion for Use with Practical Business Statistics* (2nd ed.), Irwin, 1993

Teaching

Auer has taught in the College of Business and Economics at Western Washington University from 1981 to the present. From 1975 to 1981, he taught part time for community colleges, and from 1981 to 1984 he taught part time for the Chapman College Residence Education Center System. During his career, he has taught a wide range of courses in Quantitative Methods, Production and Operations Management, Statistics, Finance and Management Information Systems. In MIS, he has taught Principles of Management Information Systems, Business Database Development, Computer Hardware and Operating Systems, and Telecommunications and Network Administration.

Education

B.A., English Literature, University of Washington, 1969
B.S., Mathematics and Economics, Western Washington University, 1978
M.A., Economics, Western Washington University, 1980
M.S., Counseling Psychology, Western Washington University, 1991

Personal

Auer is married, lives in Bellingham, Washington, and has two grown children and five grandchildren. He is active in his community, where he has been President of his neighborhood association and served on the City of Bellingham Planning and Development Commission. He enjoys music, playing acoustic and electric guitar, five-string banjo, and a bit of mandolin.