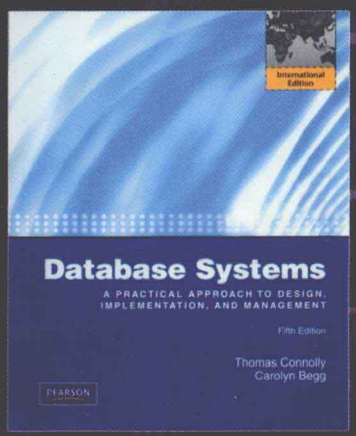
国外计算机科学教材系列

PEARSON

数据库系统
——设计、实现与管理
（第五版）

Database Systems
A Practical Approach to Design, Implementation, and Management
Fifth Edition

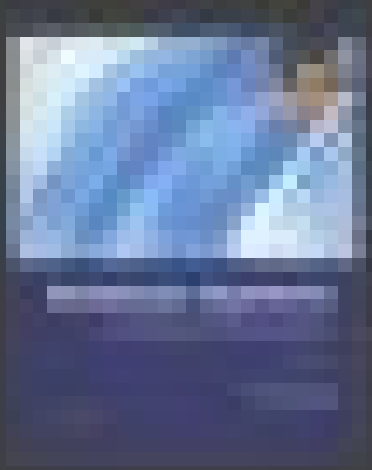


英文版

[英] Thomas M. Connolly
Carolyn E. Begg 著

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

数据库系统
——设计、实现与管理
（第五版）



国外计算机科学教材系列

数据库系统

——设计、实现与管理

（第五版）（英文版）

Database Systems

A Practical Approach to Design,

[英] Thomas M. Connolly
Carolyn E. Begg 著

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

内容简介

本书是数据库领域的经典著作，内容系统、全面、实用，被世界多所大学选为数据库相关课程的教材。全书主要内容有：数据库系统和数据库设计的基本知识；关系模型和关系语言；数据库分析和设计的主要技术；数据库设计方法学；数据库安全、事务管理、查询处理与优化；分布式DBMS与数据复制技术；面向对象数据库技术；DBMS与Web技术的结合，半结构化数据与XML；与商务智能有关的一些日益重要的技术，包括数据仓库、联机分析处理和数据挖掘以及数据库架构等。

本书既可作为计算机及相关专业本科生数据库管理或数据库设计的导论性教材（选取部分内容），也可作为研究生或本科生高年级相关课程的教材，同时亦可作为IT专业人士，如系统分析和设计人员、应用程序开发人员、系统程序员、数据库从业人员及独立的自学者的参考书。

版权贸易合同登记号 图字：01-2011-4614

图书在版编目（CIP）数据

数据库系统：设计、实现与管理 = Database Systems: A Practical Approach to Design, Implementation and Management：第5版：英文/（英）康诺利（Connolly, T. M.），（英）贝格（Begg, C. E.）著. -北京：电子工业出版社，2012.1

（国外计算机科学教材系列）

ISBN 978-7-121-14996-2

Ⅰ.①数… Ⅱ.①康… ②贝… Ⅲ.①数据库系统-研究生-教材-英文 Ⅳ.①TP311.13

中国版本图书馆CIP数据核字（2011）第227585号

策划编辑：马 岚

责任编辑：马 岚

印 刷：北京市顺义兴华印刷厂

装 订：三河市双峰印刷装订有限公司

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：70.25 字数：2338千字

印 次：2012年1月第1次印刷

定 价：109.00元

凡所购买电子工业出版社的图书有缺损问题，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

Background

The history of database research over the past 30 years is one of exceptional productivity that has led to the database system becoming arguably the most important development in the field of software engineering. The database is now the underlying framework of the information system and has fundamentally changed the way many organizations operate. In particular, the developments in this technology over the last few years have produced systems that are more powerful and more intuitive to use. This development has resulted in increasing availability of database systems for a wider variety of users. Unfortunately, the apparent simplicity of these systems has led to users creating databases and applications without the necessary knowledge to produce an effective and efficient system. And so the "software crisis" or, as it is sometimes referred to, the "software depression" continues.

The original stimulus for this book came from the authors' work in industry, providing consultancy on database design for new software systems or, as often as not, resolving inadequacies with existing systems. In addition, the authors' move to academia brought similar problems from different users—students. The objectives of this book, therefore, are to provide a textbook that introduces the theory behind databases as clearly as possible and, in particular, to provide a methodology for database design that can be used by both technical and nontechnical readers.

The methodology presented in this book for relational Database Management Systems (DBMSs)—the predominant system for business applications at present—has been tried and tested over the years in both industrial and academic environments. It consists of three main phases: conceptual, logical, and physical database design. The first phase starts with the production of a conceptual data model that is independent of all physical considerations. This model is then refined in the second phase into a logical data model by removing constructs that cannot be represented in relational systems. In the third phase, the logical data model is translated into a physical design for the target DBMS. The physical design phase considers the storage structures and access methods required for efficient and secure access to the database on secondary storage.

The methodology in each phase is presented as a series of steps. For the inexperienced designer, it is expected that the steps will be followed in the order described, and guidelines are provided throughout to help with this process. For the experienced designer, the methodology can be less prescriptive, acting more as a framework or checklist. To help the reader use the methodology and understand the important issues, the methodology has been described using a realistic worked example, based on an integrated case study, *DreamHome*. In addition, three additional case studies are provided in Appendix B to allow readers to try out the methodology for themselves.

UML (*Unified Modeling Language*)

Increasingly, companies are standardizing the way in which they model data by selecting a particular approach to data modeling and using it throughout their database development projects. A popular high-level data model used in conceptual/ logical database design, and the one we use in this book, is based on the concepts of the Entity–Relationship (ER) model. Currently there is no standard notation for an ER model. Most books that cover database design for relational DBMSs tend to use one of two conventional notations:

- Chen's notation, consisting of rectangles representing entities and diamonds representing relationships, with lines linking the rectangles and diamonds; or
- Crow's Feet notation, again consisting of rectangles representing entities and lines between entities representing relationships, with a crow's foot at one end of a line representing a one-to-many relationship.

Both notations are well supported by current Computer-Aided Software Engineering (CASE) tools. However, they can be quite cumbersome to use and a bit difficult to explain. In previous editions, we used Chen's notation. However, following an extensive questionnaire carried out by Pearson Education, there was a general consensus that the notation should be changed to the latest objectoriented modeling language, called UML (Unified Modeling Language). UML is a notation that combines elements from the three major strands of object-oriented design: Rumbaugh's OMT modeling, Booch's Object-Oriented Analysis and Design, and Jacobson's Objectory.

There are three primary reasons for adopting a different notation: (1) UML is becoming an industry standard; for example, the Object Management Group (OMG) has adopted UML as the standard notation for object methods; (2) UML is arguably clearer and easier to use; (3) UML is now being adopted within academia for teaching object-oriented analysis and design, and using UML in database modules provides more synergy. Therefore, in this edition we have adopted the class diagram notation from UML. We believe that you will find this notation easier to understand and use.

What's New in the Fifth Edition

- New chapter on database architectures and the Web, covering *n*-tier architectures, Web services, and service-oriented architecture (SOA).
- New chapter on professional, legal, and ethical issues in IT and databases.
- New chapter on advanced SQL, including coverage of the SQL programming language (SQL/PSM), stored procedures, and triggers.
- Updated treatment to cover the latest version of the SQL standard, which was released in summer 2008 (SQL:2008).
- Extended treatment of Web-DBMS integration, including coverage of the Java™ persistence mechanism, Java Persistence API (JPA), as well as updated treatment of the latest version of some of the other Java persistence mechanisms, plus an updated section covering .NET 3.5.
- Extended treatment of XML, SPARQL, XQuery 1.0 and XPath 2.0 (including the new XQuery Update facility), and the new SQL:2008 SQL/XML standard.

- Extended treatment of data warehousing, OLAP, and data mining, including the SQL:2008 functionality in these areas.
- Coverage updated to Oracle11*i* and Microsoft Office Access® 2007.

Intended Audience

This book is intended as a textbook for a oneor two-semester course in database management or database design in an introductory undergraduate, graduate, or advanced undergraduate course. Such courses are usually required in an information systems, business IT, or computer science curriculum.

The book is also intended as a reference book for IT professionals, such as systems analysts or designers, application programmers, systems programmers, database practitioners, and for independent self-teachers. Owing to the widespread use of database systems nowadays, these professionals could come from any type of company that requires a database.

It would be helpful for students to have a good background in the file organization and data structures concepts covered in Appendix F before covering the material in Chapter 18 on physical database design and Chapter 23 on query processing. This background ideally will have been obtained from a prior course. If this is not possible, then the material in Appendix F can be presented near the beginning of the database course, immediately following Chapter 1.

An understanding of a high-level programming language, such as C, would be advantageous for Appendix I on embedded and dynamic SQL and Section 28.3 on ObjectStore.

Distinguishing Features

(1) An easy-to-use, step-by-step methodology for conceptual and logical database design, based on the widely accepted Entity–Relationship model, with normalization used as a validation technique. There is an integrated case study showing how to use the methodology.

(2) An easy-to-use, step-by-step methodology for physical database design, covering the mapping of the logical design to a physical implementation, the selection of file organizations and indexes appropriate for the applications, and when to introduce controlled redundancy. Again, there is an integrated case study showing how to use the methodology.

(3) Separate chapters showing how database design fits into the overall database systems development lifecycle, how fact-finding techniques can be used to identify the system requirements, and how UML fits into the methodology.

(4) A clear and easy-to-understand presentation, with definitions clearly highlighted, chapter objectives clearly stated, and chapters summarized. Numerous examples and diagrams are provided throughout each chapter to illustrate the concepts. There is a realistic case study integrated throughout the book and additional case studies that can be used as student projects.

(5) Extensive treatment of the latest formal and de facto standards: Structured Query Language (SQL), Query-By-Example (QBE), and the Object Data Management Group (ODMG) standard for object-oriented databases.

(6) Three tutorial-style chapters on the SQL standard, covering both interactive and embedded SQL.

(7) A chapter on legal, professional and ethical issues related to IT and databases.

(8) Comprehensive coverage of the concepts and issues relating to distributed DBMSs and replication servers.

(9) Comprehensive introduction to the concepts and issues relating to object-based DBMSs including a review of the ODMG standard and a tutorial on the object management facilities within the latest release of the SQL standard, SQL:2008.

(10) Extensive treatment of the Web as a platform for database applications with many code samples of accessing databases on the Web. In particular, we cover persistence through Container-Managed Persistence (CMP), Java Data Objects (JDO), Java Persistence API (JPA), JDBC, SQLJ, ActiveX Data Objects (ADO), ADO.NET, and Oracle PL/SQL Pages (PSP).

(11) An introduction to semistructured data and its relationship to XML and extensive coverage of XML and its related technologies. In particular, we cover XML Schema, XQuery, and the XQuery Data Model and Formal Semantics. We also cover the integration of XML into databases and examine the extensions added to SQL:2003 and SQL:2008 to enable the publication of XML.

(12) Comprehensive introduction to data warehousing, Online Analytical Processing (OLAP), and data mining.

(13) Comprehensive introduction to dimensionality modeling for designing a data warehouse database. An integrated case study is used to demonstrate a methodology for data warehouse database design.

(14) Coverage of DBMS system implementation concepts, including concurrency and recovery control, security, and query processing and query optimization.

Pedagogy

Before starting to write any material for this book, one of the objectives was to produce a textbook that would be easy for the readers to follow and understand, whatever their background and experience. From the authors' experience of using textbooks, which was quite considerable before undertaking a project of this size, and also from listening to colleagues, clients, and students, we knew there were a number of design features that readers liked and disliked. With these comments in mind, the following style and structure features were adopted:

- A set of objectives is clearly identified at the start of each chapter.
- Each important concept that is introduced is clearly defined and highlighted by setting the definition apart from the text.
- Diagrams are liberally used throughout to support and clarify concepts.
- A very practical orientation: each chapter contains many worked examples to illustrate the concepts covered.
- A summary at the end of each chapter covers the main concepts introduced.
- A set of review questions falls at the end of each chapter, the answers to which can be found in the text.
- A set of exercises at the end of each chapter, can be used by teachers or by individuals to demonstrate and test the individual's understanding of the chapter, the answers to which can be found in the accompanying Instructor's Solutions Manual.

Support Materials

A comprehensive set of supplements are available for this textbook:

- Lecture slides in PowerPoint® format
- Instructor's Solutions Manual, including sample solutions to all review questions and exercises
- A companion website with additional resources, located at: `http://www.aw.com/connollybegg`

Supplements are available to qualified instructors only at `www.pearsonhighered.com/irc`. Please contact your local sales representative or send e-mail to `computing@aw.com` for access information.

Organization of this Book

Part 1: Background

Part 1 of the book serves to introduce the field of database systems and database design.

Chapter 1 introduces the field of database management, examining the problems with the precursor to the database system, the file-based system, and the advantages offered by the database approach.

Chapter 2 examines the database environment, discussing the advantages offered by the three-level ANSI-SPARC architecture, introducing the most popular data models and outlining the functions that should be provided by a multi-user DBMS.

Chapter 3 examines multi-user DBMS architectures and discusses the different types of middleware that exist in the database field. It also examines Web services that can be used to provide new types of business services to users and service-oriented architecture (SOA). The chapter briefly introduces the architecture for a distributed DBMS and data warehousing presented more fully in later chapters. The chapter also looks at the underlying software architecture for DBMSs and the logical and physical structures in the Oracle DBMS, which could be omitted for a first course in database management.

Part 2: The Relational Model and Languages

Part 2 of the book serves to introduce the relational model and relational languages, namely the relational algebra and relational calculus, QBE (Query-By-Example), and SQL (Structured Query Language). This part also examines two highly popular commercial systems: Microsoft Office Access and Oracle.

Chapter 4 introduces the concepts behind the relational model, the most popular data model at present, and the one most often chosen for standard business applications. After introducing the terminology and showing the relationship with mathematical relations, the relational integrity rules, entity integrity, and referential integrity are discussed. The chapter concludes with a section on views, which is expanded upon in Chapter 7.

Chapter 5 introduces the relational algebra and relational calculus with examples to illustrate all the operations. This could be omitted for a first course in database management. However, relational algebra is required to understand Query Processing in Chapter 23 and fragmentation in Chapter 24 on distributed DBMSs. In addition, the comparative aspects of the procedural algebra and the non-procedural calculus act as a useful precursor for the study of SQL in Chapters 6 and 7, although not essential.

Chapter 6 introduces the data manipulation statements of the SQL standard: SELECT, INSERT, UPDATE, and DELETE. The chapter is presented as a tutorial, giving a series of worked examples that demonstrate the main concepts of these statements.

Chapter 7 covers the main data definition facilities of the SQL standard. Again, the chapter is presented as a worked tutorial. The chapter introduces the SQL data types and the data definition statements, the Integrity Enhancement Feature (IEF), and the more advanced features of the data definition statements, including the access control statements GRANT and REVOKE. It also examines views and how they can be created in SQL.

Chapter 8 covers some of the advanced features of SQL, including the SQL programming language (SQL/PSM), triggers, and stored procedures.

Chapter 9 is another practical chapter that examines the interactive query language, Query-By-Example (QBE), which has acquired the reputation of being one of the easiest ways for nontechnical computer users to access information in a database. The QBE language is demonstrated using Microsoft Office Access.

Part 3: Database Analysis and Design Techniques

Part 3 of the book discusses the main techniques for database analysis and design and how they can be applied in a practical way.

Chapter 10 presents an overview of the main stages of the database system development lifecycle. In particular, it emphasizes the importance of database design and shows how the process can be decomposed into three phases: conceptual, logical, and physical database design. It also describes how the design of the application (*the functional approach*) affects database design (*the data approach*). A crucial stage in the database system development lifecycle is the selection of an appropriate DBMS. This chapter discusses the process of DBMS selection and provides some guidelines and recommendations.

Chapter 11 discusses when a database developer might use fact-finding techniques and what types of facts should be captured. The chapter describes the most commonly used fact-finding techniques and identifies the advantages and disadvantages of each. The chapter also demonstrates how some of these techniques may be used during the earlier stages of the database system lifecycle using the *DreamHome* case study.

Chapters 12 and 13 cover the concepts of the Entity–Relationship (ER) model and the Enhanced Entity–Relationship (EER) model, which allows more advanced data modeling using subclasses and superclasses and categorization. The EER model is a popular high-level conceptual data model and is a fundamental technique of the database design methodology presented herein. The reader is also introduced to UML to represent ER diagrams.

Chapters 14 and 15 examine the concepts behind normalization, which is another important technique used in the logical database design methodology. Using a series of worked examples drawn from the integrated case study, they demonstrate how to transition a design from one normal form to another and show the advantages of having a logical database design that conforms to particular normal forms up to and including fifth normal form.

Part 4: Methodology

This part of the book covers a methodology for database design. The methodology is divided into three parts covering conceptual, logical, and physical database design. Each part of the methodology is illustrated using the *DreamHome* case study.

Chapter 16 presents a step-by-step methodology for conceptual database design. It shows how to decompose the design into more manageable areas based on individual user views, and then provides guidelines for identifying entities, attributes, relationships, and keys.

Chapter 17 presents a step-by-step methodology for logical database design for the relational model. It shows how to map a conceptual data model to a logical data model and how to validate it to ensure that it supports the required transactions and follows the rules of normalization. For database systems with multiple user views, this chapter shows how to merge the resulting local data models together into a global data model that represents all the user views of the part of the enterprise being modeled.

Chapters 18 and 19 present a step-by-step methodology for physical database design for relational systems. It shows how to translate the logical data model developed during logical database design into a physical design for a relational system. The methodology addresses the performance of the resulting implementation by providing guidelines for choosing file organizations and storage structures and when to introduce controlled redundancy.

Part 5: Selected Database Issues

Part 5 of the book examines four specific topics that the authors consider necessary for a modern course in database management.

Chapter 20 considers database security and administration. Security considers both the DBMS and its environment. It illustrates security provision with Microsoft Office Access and Oracle. The chapter also examines the security problems that can arise in a Web environment and presents some approaches to overcoming them. The chapter concludes with a discussion of the tasks of data administration and database administration.

Chapter 21 considers professional, legal, and ethical issues related to IT and data management. It distinguishes between legal and ethical issues and situations that data/database administrators face, discusses how new regulations are placing additional requirements and responsibilities on data/database administrators, and how legislation, such as the Sarbanes-Oxley Act and the Basel II accords, affect data/database administration functions.

Chapter 22 concentrates on three functions that a Database Management System should provide, namely transaction management, concurrency control, and recovery. These functions are intended to ensure that the database is reliable and remains in a consistent state when multiple users are accessing the database and in the presence of failures of both hardware and software components. The chapter also discusses advanced transaction models that are more appropriate for transactions that may be of a long duration. The chapter concludes by examining transaction management within Oracle.

Chapter 23 examines query processing and query optimization. The chapter considers the two main techniques for query optimization: the use of heuristic rules that order the operations in a query and the other technique that compares different strategies based on their relative costs and selects the one that minimizes resource usage. The chapter concludes by examining query processing within Oracle.

Part 6: Distributed DBMSs and Replication

Part 6 of the book examines distributed DBMSs and object-based DBMSs. Distributed database management system (DDMS) technology is one of the current major developments in the database systems area. The previous chapters of this book concentrate on centralized database systems, that is, systems with a single logical database located at one site under the control of a single DBMS.

Chapter 24 discusses the concepts and problems of distributed DBMSs, with which users can access the database at their own site, and also access data stored at remote sites.

Chapter 25 examines various advanced concepts associated with distributed DBMSs. In particular, it concentrates on the protocols associated with distributed transaction management, concurrency control, deadlock management, and database recovery. The chapter also examines the X/Open Distributed Transaction Processing (DTP) protocol. The chapter concludes by examining data distribution within Oracle.

Chapter 26 discusses replication servers as an alternative to distributed DBMSs and examines the issues associated with mobile databases. The chapter also examines the data replication facilities in Oracle.

Part 7: Object DBMSs

The preceding chapters of this book concentrate on the relational model and relational systems. The justification for this is that such systems are currently the predominant DBMS for traditional business database applications. However, relational systems are not without their failings, and the object-based DBMS is a major development in the database systems area that attempts to overcome these failings. Chapters 27–29 examine this development in some detail.

Chapter 27 introduces the object-based DBMSs and first examines the types of advanced database applications that are emerging and discusses the weaknesses of the relational data model that makes it unsuitable for these types of applications. It then examines the object-oriented DBMS (OODBMS) and starts by providing an introduction to object-oriented data models and persistent programming languages. The chapter discusses the difference between the two-level storage model used by conventional DBMSs and the single-level model used by OODBMSs and how this affects data access. It also discusses the various approaches to providing persistence in programming languages and the different techniques for pointer swizzling and examines version management, schema evolution, and OODBMS architectures. The chapter concludes by briefly showing how the methodology presented in Part 4 of this book may be extended for object-oriented databases.

Chapter 28 addresses the object model proposed by the Object Data Management Group (ODMG), which has become a de facto standard for OODBMSs. The chapter also examines ObjectStore, a commercial OODBMS.

Chapter 29 examines the object-relational DBMS and provides a detailed overview of the object management features that have been added to the new release of the SQL standard, SQL:2008. The chapter also discusses how query processing and query optimization need to be extended to handle data type extensibility efficiently. The chapter concludes by examining some of the object-relational features within Oracle.

Part 8: Web and DBMSs

Part 8 of the book deals with the integration of the DBMS into the Web environment, semistructured data and its relationship to XML, XML query languages, and mapping XML to databases.

Chapter 30 examines the integration of the DBMS into the Web environment. After providing a brief introduction to Internet and Web technology, the chapter examines the appropriateness of the Web as a database application platform and discusses the advantages and disadvantages of this approach. It then considers a number of the different approaches to integrating DBMSs into the Web environment, including scripting languages, CGI, server extensions, Java, ADO and ADO.NET, and Oracle's Internet Platform.

Chapter 31 examines semistructured data and then discusses XML and how XML is an emerging standard for data representation and interchange on the Web. The chapter then discusses XML-related technologies such as namespaces, XSL, XPath, XPointer, XLink, SOAP, WSDL, and UDDI. It also examines how XML Schema can be used to define the content model of an XML document and how the Resource Description Framework (RDF) provides a framework for the exchange of metadata. The chapter examines query languages for XML and, in particular, concentrates on XQuery, as proposed by W3C. It also examines the extensions added to SQL:2008 to enable the publication of XML and, more generally, mapping and storing XML in databases.

Part 9: Business Intelligence

The final part of the book considers the main technologies associated with Business Intelligence (BI): the data warehouse, Online Analytical Processing (OLAP), and data mining.

Chapter 32 discusses data warehousing, what it is, how it has evolved, and describes the potential benefits and problems associated with this system. The chapter examines the architecture, the main components, and the associated tools and technologies of a data warehouse. The chapter also discusses data marts and the issues associated with the development and management of data marts. The chapter concludes by describing the data warehousing facilities of the Oracle DBMS.

Chapter 33 describes alternative approaches for the development of the decisionsupport database of a data warehouse or data mart. The chapter describes the basic concepts associated with dimensionality modeling and compares this technique with traditional Entity–Relationship (ER) modeling. It also describes and demonstrates a step-by-step methodology for designing a data warehouse using worked examples taken from an extended version of the *DreamHome* case study. The chapter concludes by describing how to design a data warehouse using the Oracle Warehouse Builder.

Chapter 34 describes Online Analytical Processing (OLAP). It discusses what OLAP is and the main features of OLAP applications. The chapter discusses how multidimensional data can be represented and the main categories of OLAP tools. It also discusses the OLAP extensions to the SQL standard and how Oracle supports OLAP.

Chapter 35 describes data mining (DM). It discusses what DM is and the main features of DM applications. The chapter describes the main characteristics of data mining operations and associated techniques. It describes the process of DM and the main features of DM tools, with particular coverage of Oracle DM.

Appendices — In Book

Appendix A provides a description of *DreamHome*, a case study that is used extensively throughout the book.

Appendix B provides three additional case studies, which can be used as student projects.

Appendix C describes two alternative data modeling notations to UML: Chen's notation and Crow's Foot.

Appendix D summarizes the steps in the methodology presented in Chapters 16–19 for conceptual, logical, and physical database design.

Appendix E examines an open-source lightweight RDBMS called Pyrrho implemented in C# that demonstrates many of the concepts discussed in this book and can be downloaded and used.

Appendices—Online, at http://www.aw.com/connollybegg

Appendix F provides some background information on file organization and storage structures that is necessary for an understanding of the physical database design methodology presented in Chapter 18 and query processing in Chapter 23.

Appendix G describes Codd's 12 rules for a relational DBMS, which form a yardstick against which the "real" relational DBMS products can be identified.

Appendix H provides introductions to two popular commercial relational DBMSs: Microsoft Office Access and Oracle. Elsewhere in the book, we examine how these systems implement various database facilities, such as security and query processing.

Appendix I examines embedded and dynamic SQL, with sample programs in C. The chapter also examines the Open Database Connectivity (ODBC) standard, which has emerged as a de facto industry standard for accessing heterogeneous SQL databases.

Appendix J discusses how to estimate the disk space requirements for an Oracle database.

Appendix K provides an overview of the main object-oriented concepts.

Appendix L provides some sample Web scripts to complement Chapter 30 on Web technology and DBMSs.

The logical organization of the book and the suggested paths through it are illustrated in Figure P.1.

Corrections and Suggestions

As a textbook of this size is vulnerable to errors, disagreements, omissions, and confusion; your input is solicited for future reprints and editions. Comments, corrections, and constructive suggestions should be sent to me, at:

thomas.connolly@uws.ac.uk

Acknowledgments

This book is the outcome of our many years of work in industry, research, and academia. It is difficult to name everyone who has helped us in our efforts. We apologize to anyone we may happen to omit. We first give special thanks and apologies to our families, who over the years have been neglected, even ignored, during our deepest concentrations.

We would like to thank the reviewers of the previous editions of the book: William H. Gwinn, Texas Tech University; Adrian Larner, De Montfort University, Leicester; Andrew McGettrick, University of Strathclyde; Dennis McLeod, University of Southern California; Josephine DeGuzman Mendoza, California State University; Jeff Naughton, University of Oklahoma; Junping Sun, Nova Southeastern University; Donovan Young, Georgia Tech; Barry Eaglestone, University of Bradford; John Wade, IBM; Stephano Ceri, Politecnico di Milano; Lars Gillberg, Mid Sweden University, Oestersund; Dawn Jutla, St Mary's University, Halifax; Julie McCann, City University, London; Munindar Singh, North Carolina State University; Hugh Darwen, Hursely, UK; Claude Delobel, Paris, France; Dennis Murray, Reading, UK; Richard Cooper, University of Glasgow; Emma Eliason, University of Orebro; Sari Hakkarainen, Stockholm University and the Royal Institute of Technology; Nenad Jukic, Loyola University Chicago; Jan Paredaens, University of Antwerp; Stephen Priest, Daniel Webster College; and from our own department, John Kawala and Peter Knaggs. Many others are still anonymous to us—we thank you for the time you must have spent on the manuscript. We would also like to acknowledge Anne Strachan for her contribution to the first edition.

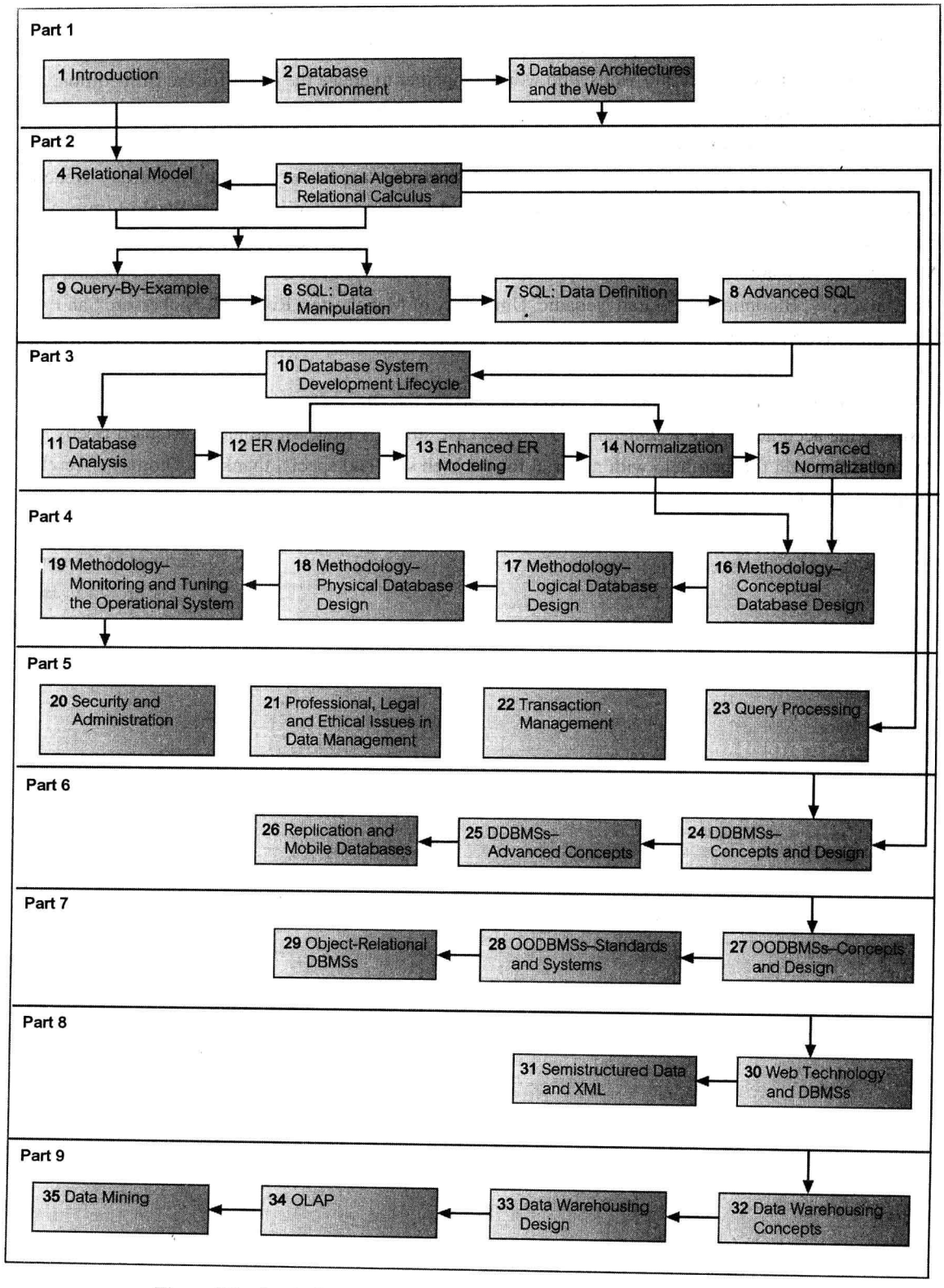


Figure P.1 Logical organization of the book and suggested paths through it.