国外计算机科学教材系列

数据库系统

——设计、实现与管理（第三版）

Database Systems

A Practical Approach to Design, Implementation, and Management

Third Edition

英文版

［英］ Thomas Connolly
Carolyn Begg 著

PEARSON
Addison Wesley

PHEI
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

数据库系统
——设计、实现与管理
（第三版）

（英文版）

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

［英］ Thomas Connolly
Carolyn 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课程的教材。全书共分八个部分，其中的主要内容有：数据库系统和数据库设计的基础知识；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、询问处理与优化和嵌入式SQL；分布式DBMS和基于对象的DBMS，以及一些日益重要的技术，包括DBMS与Web技术的连接、半结构化技术与XML的关系、数据仓库、联机分析处理和数据挖掘等。

本书可作为计算机科学与技术及相关专业高年级本科生及研究生的教材，同时也可作为数据库研究和开发人员的参考书。

版权贸易合同登记号：图字：01-2002-4137

图书在版编目（CIP）数据

数据库系统：设计、实现与管理，第三版 = Database Systems: A Practical Approach to Design, Implementation, and Management, Third Edition/（英）康诺利（Connolly, T.）等著. -北京：电子工业出版社，2003.6
（国外计算机科学教材系列）
ISBN 7-5053-8784-7

Ⅰ.数… Ⅱ.康… Ⅲ.数据库系统-教材-英文 Ⅳ. TP311.13

中国版本图书馆CIP数据核字（2003）第043620号

责任编辑：杜　萌
印 刷 者：北京兴华印刷厂
出版发行：电子工业出版社　http://www.phei.com.cn
　　　　　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：61.75　字数：1383千字
版　　次：2003年6月第1版　　2003年6月第1次印刷
定　　价：79.00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

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 has resulted in database systems becoming increasingly available to 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. Added to this, 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 non-technical 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 considera-tions. This model is then refined in the second phase into a logical data model by remov-ing 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 guide-lines 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 represent-ing 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 CASE tools. However, they can be quite cumbersome to use and a bit difficult to explain. Prior to this edition, 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 object-oriented 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 becom-ing an industry standard; for example, the Object Management Group (OMG) has adopted the 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 you will find this notation easier to understand and use. Prior to making this move to UML, we spent a considerable amount of time experimenting with UML and checking its suitability for database design. We concluded this work by publishing a book through Pearson Education called *Database Solutions: A Step-by-Step Guide to Building Databases*. This book uses the methodology to design and build databases for two case studies, one with the target DBMS as Microsoft Access and one with the target database as Oracle. This book also contains many other case studies with sample solutions, and an accompanying CD-ROM with an evaluation copy of the visual modeling tool Rational Rose.

What's New in the Third Edition

The second edition of the book has been revised to improve readability, to update or to extend coverage of existing material, and to include new material. The major changes in the third edition are as follows.

- New chapter covering the relational algebra and relational calculus.
- New chapter providing an introduction to two of the most popular commercial relational DBMSs, Microsoft Access and Oracle.
- New chapter on fact-finding techniques and how these techniques can be used during the database application lifecycle particularly in the early stages, such as database planning, system definition, and requirements collection and analysis.
- Coverage of the Entity–Relationship (ER) and the Enhanced Entity–Relationship (EER) model using the class diagram notation from UML.
- Improvements to the database design methodology. Restructuring of the methodology into four chapters, with an integrated demonstration of the methodology in practice using the *DreamHome* case study.
- Restructuring of the previous two SQL tutorial chapters into three chapters: one cover-ing data manipulation, one covering data definition, and one covering embedded SQL and the Open Database Connectivity (ODBC) standard.
- Extended chapter on security to cover security in Microsoft Access and Oracle, and security in a Web environment.
- Coverage of transaction management is extended to examine transaction management in Oracle.

- Coverage of query processing is extended to examine query processing in Oracle.
- Coverage of distributed DBMSs and replication servers is extended to examine data distribution and replication within Oracle.
- The chapter on Object-Oriented DBMSs (OODBMSs) is divided into two chapters. The first chapter discusses general issues relating to OODBMSs and persistent programming languages. The second chapter examines the new 1999 Object Data Management Group (ODMG) standard and provides an overview of ObjectStore, a commercial OODBMS.
- A revised tutorial chapter on Object-Relational DBMSs covering the formal release ofthe SQL3 standard.
- A revised chapter on Web technology and DBMSs. An appendix providing many programming examples of integrating databases into a Web environment covering JavaScript, PHP and PostgresQL, CGI and Perl, Java database connectivity using JDBC and SQLJ, JavaServer Pages, Active Server Pages, and Oracle's PL/SQL Server Pages.
- A new chapter on semistructured data and XML, its related technologies, and the some proposals for a query language for XML.
- The chapter on data warehousing is divided into two. The first chapter introduces data warehousing and the second chapter covers data warehousing database design using dimensionality modeling.

Intended Audience

This book is intended as a textbook for a one- or two-semester course in database manage-ment or database design in an introductory undergraduate course, a graduate or advanced undergraduate course. Such courses are usually required in an information systems, busi-ness 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 practi-tioners, and for independent self-teachers. Owing to the widespread use of database sys-tems 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 C before covering the material in Chapter 16 on physical database design and Chapter 20 on query processing. This background ideally will have been obtained from a prior course. If this is not possible, then the material in Appendix C 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 advan-tageous for Chapter 21 on embedded and dynamic SQL and Section 26.3 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) There are separate chapters showing how database design fits into the overall sys-tems development lifecycle and how fact-finding techniques can be used to identify the system requirements.

(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 further case studies that can be used as student projects.

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

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

(7) An overview chapter covering two of the most popular commercial DBMSs: Micro-soft Access and Oracle. Many of the subsequent chapters examine how Microsoft Access and Oracle support the mechanisms that are being discussed.

(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 the increasingly important area of object-based DBMSs including a review of the ODMG standard, and a tutorial on the object management facilities that have been added to the latest release of the SQL standard, SQL3.

(10) Extensive treatment of the Web as a platform for database applications with many code samples of accessing databases on the Web.

(11) An introduction to semistructured data, its relationship to XML, and proposals for a query language for XML.

(12) Comprehensive introduction to data warehousing, online analytical processing (OLAP), and data mining.

(13) Comprehensive introduction to dimensionality modeling for designing a data ware-house 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, there were a number of design features that readers liked and disliked. With these comments in mind, the following style and structure was adopted:

- A set of objectives, clearly identified at the start of each chapter.
- Each important concept that is introduced is clearly defined and highlighted by placing the definition in a box.
- Diagrams are liberally used throughout to support and clarify concepts.
- A very practical orientation: to this end, each chapter contains many worked examples to illustrate the concepts covered.
- A summary at the end of each chapter, covering the main concepts introduced.
- A set of review questions, the answers to which can be found in the text.
- A set of exercises that 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 Guide.

Instructor's Guide

A comprehensive supplement containing numerous instructional resources is available for this textbook, upon request to Pearson Education. The accompanying Instructor's Guide includes:

- *Course structures* These include suggestions for the material to be covered in a variety of courses.

- *Teaching suggestions* These include lecture suggestions, teaching hints, and student project ideas that make use of the chapter content.
- *Solutions* Sample answers are provided for all review questions and exercises.
- *Examination questions* Examination questions (similar to the questions and exercises at the end of each chapter), with solutions.
- *Transparency masters* An electronic set of overhead transparencies containing the main points from each chapter, enlarged illustrations and tables from the text, help the instructor to associate lectures and class discussion to material in the textbook.

Additional information about the Instructor's Guide and the book can be found on thePearson Education Web site at:

http://www.booksites.net/connolly

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. The chapter also looks at the underlying software architecture for DBMSs, 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 Access and Oracle.

Chapter 3 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 an overview on views, which is expanded upon in Chapter 6.

Chapter 4 introduces the relational algebra and relational calculus with examples to illus-trate 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 20 and fragmentation in Chapter 22 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 5 and 6, although not essential.

Chapter 5 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 6 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 7 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 non-technical computer users to access information in a database. QBE is demon-strated using Microsoft Access.

Chapter 8 completes the second part of the book by providing introductions to two popular commercial relational DBMSs, namely Microsoft Access and Oracle. In sub-sequent chapters of the book, we examine how these systems implement various database facilities, such as security and query processing.

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 9 presents an overview of the main stages of the database application 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 application lifecycle is the selection of an appropriate DBMS. This chapter discusses the process of DBMS selection and provides some guidelines and recommendations. The chapter concludes with a dis-cussion of the importance of data administration and database administration.

Chapter 10 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 application lifecycle using the *DreamHome* case study.

Chapter 11 and 12 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 meth-odology presented herein. The reader is also introduced to UML to represent ER diagrams.

Chapter 13 examines 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, it demonstrates how to transition a design from one normal form to another and shows 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 14 presents a step-by-step methodology for conceptual database design. It shows how to decompose the design into more manageable areas based on individual views, and then provides guidelines for identifying entities, attributes, relationships, and keys.

Chapter 15 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 against the required transactions using the technique of normaliza-tion. For database applications with multiple user views, this chapter shows how to *merge* the resulting data models together into a global data model that represents all the views of the part of the enterprise being modeled.

Chapter 16 and 17 present a step-by-step methodology for physical database design for relational systems. It shows how to translate the global 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 by considering denormalization —— the introduction of 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 18 considers database security, not just in the context of DBMS security but also in the context of the security of the DBMS environment. It illustrates security provision with Microsoft Access and Oracle. The chapter also examines the security problems that can arise in a Web environment and presents some approaches to overcoming them.

Chapter 19 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 20 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 con-cludes by examining query processing within Oracle.

Chapter 21 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.

Part 6 Current Trends

Part 6 of the book examines distributed DBMSs and object-based DBMSs. Distributed database management system (DDBMS) technology is one of the current major develop-ments 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 22 discusses the concepts and problems of distributed DBMSs, where users can access the database at their own site and also access data stored at remote sites.

Chapter 23 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, and discusses replication servers as an alternative to distributed DBMSs. The chapter concludes by examining data distribution and replication within Oracle.

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 24–27 examine this development in some detail.

Chapter 24 acts as an introduction to 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. The chapter then introduces the main concepts of object orientation. It also discusses the problems of storing objects in a relational database.

Chapter 25 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 lan-guages and the different techniques for pointer swizzling, and examines version manage-ment, 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 26 addresses the new 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 27 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 stan-dard, SQL3. 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 7 Emerging Trends

The final part of the book deals with five emerging areas that are becoming particularly important, namely, the integration of the DBMS into the Web environment, semistructured data and its relationship to XML, data warehousing, online analytical processing (OLAP), and data mining.

Chapter 28 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 lan-guages, CGI, server extensions, Java, Active Server Pages, and Oracle's Internet Platform.

Chapter 29 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, and XLink. 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 frame-work for the exchange of meta-data. The chapter concludes by examining query languages for XML.

Chapter 30 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 31 provides an approach to the design of the database of a data warehouse/ data mart built to support decision-making. 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 32 deals with online analytical processing (OLAP) and data mining. The chapter discusses the concepts associated with multi-dimensional databases and highlights the characteristics of the three main OLAP tools, namely multi-dimensional (MOLAP), Relational OLAP (ROLAP), and Managed Query Environment (MQE) tools. It also examines how SQL has been extended to provide complex data analysis functions using, as an example, Red Bricks Intelligent SQL (RISQL). The chapter then presents the con-cepts associated with data mining and identifies the main characteristics of data mining operations, techniques, and tools, and examines the relationship between data mining and data warehousing.

Appendices

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 provides some background information on file organization and storage structures that is necessary for an understanding of the physical database design method-ology presented in Chapter 16 and Query Processing in Chapter 20.

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

Appendix E describes two alternative data modeling notations to UML, namely Chen's notation and Crow's Foot.

Appendix F summarizes the steps in the methodology presented in Chapters 14–17 for conceptual, logical, and physical database design.

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

Appendix H provides some sample Web scripts to complement Chapter 28 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 so 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 Pearson Education, or by electronic mail to:

thomas.connolly@paisley.ac.uk

Acknowledgments

This book is the outcome of many years of work by the authors in industry, research, and academia. It is therefore difficult to name all the people who have directly or indirectly helped us in our efforts; an idea here and there may have appeared insignificant at the time but may have had a significant causal effect. For those people we are about to omit, we apologize now. However, special thanks and apologies must first go to our families, who over the years have been neglected, even ignored, during our deepest concentrations.

Next, for the first edition, we should like to thank our editors Dr Simon Plumtree and Nicky Jaeger, for their help, encouragement, and professionalism throughout this time; and our production editor Martin Tytler, and copy editor Lionel Browne. We should also like to thank the reviewers of the first edition, who contributed their comments, sugges-tions, and advice. In particular, we would like to mention: William H. Gwinn, Instructor, Texas Tech University; Adrian Larner, De Montfort University, Leicester; Professor Andrew McGettrick, University of Strathclyde; Dennis McLeod, Professor of Computer Science, University of Southern California; Josephine DeGuzman Mendoza, Associate Professor, California State University; Jeff Naughton, Professor A. B. Schwarzkopf, University of Oklahoma; Junping Sun, Assistant Professor, Nova Southeastern University; Donovan Young, Associate Professor, Georgia Tech; Dr Barry Eaglestone, Lecturer in Computer Science, University of Bradford; John Wade, IBM. We would also like to acknowledge Anne Strachan for her contribution to the first edition.

For the second edition, we would first like to thank Sally Mortimore, our editor, and Martin Klopstock and Dylan Reisenberger in the production team. We should also like to thank the reviewers of the second edition, who contrib-

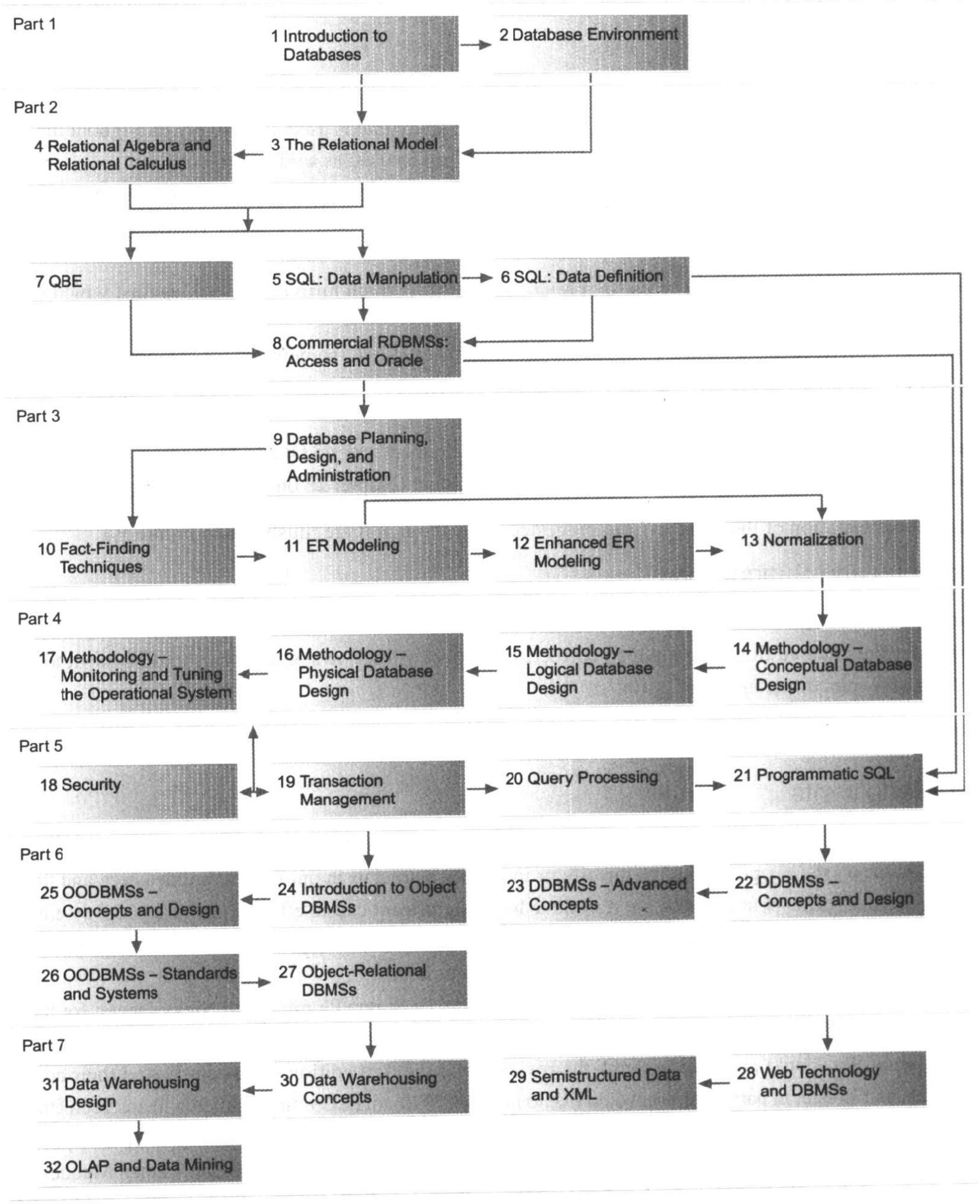
Part 1
1 Introduction to Databases
2 Database Environment
Part 2
4 Relational Algebra and Relational Calculus
3 The Relational Model
7 QBE
5 SQL: Data Manipulation
6 SQL: Data Definition
8 Commercial RDBMSs: Access and Oracle
Part 3
9 Database Planning, Design, and Administration
10 Fact-Finding Techniques
11 ER Modeling
12 Enhanced ER Modeling
13 Normalization
Part 4
17 Methodology – Monitoring and Tuning the Operational System
16 Methodology – Physical Database Design
15 Methodology – Logical Database Design
14 Methodology – Conceptual Database Design
Part 5
18 Security
19 Transaction Management
20 Query Processing
21 Programmatic SQL
Part 6
25 OODBMSs – Concepts and Design
24 Introduction to Object DBMSs
23 DDBMSs – Advanced Concepts
22 DDBMSs – Concepts and Design
26 OODBMSs – Standards and Systems
27 Object-Relational DBMSs
Part 7
31 Data Warehousing Design
30 Data Warehousing Concepts
29 Semistructured Data and XML
28 Web Technology and DBMSs
32 OLAP and Data Mining

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

uted their comments, suggestions, and advice. In particular, we would like to mention: Stephano Ceri, Politecnico di Milano; Lars Gillberg, Mid Sweden University, Oestersund; Dawn Jutla, St Mary's University, Halifax, Canada; Julie McCann, City University, London; Munindar Singh, North Carolina State University; Hugh Darwen, Hursely, UK; Claude Delobel, Paris, France; Dennis Murray, Reading, UK; and from our own department John Kawala and Dr Peter Knaggs.

For the third edition, we would first like to thank Kate Brewin, our editor, and Stuart Hay and Mary Lince in the production team. We should also like to thank the reviewers of the second edition, who contributed their comments, suggestions, and advice. In par-ticular, we would like to mention: Richard Cooper, University of Glasgow, UK; Emma Eliason, University of Orebro, Sweden; Sari Hakkarainen, Stockholm University and the Royal Institute of Technology; Nenad Jukic, Loyola University Chicago, USA; Jan Paredaens, University of Antwerp, Belgium; Stephen Priest, Daniel Webster College, USA. Many others are still anonymous to us —— we thank you for the time you must have spent on the manuscript.

We should also like to thank Malcolm Bronte-Stewart for the *DreamHome* concept, Moira O'Donnell for ensuring the accuracy of the *Wellmeadows Hospital* case study, and special thanks to Thomas's secretary Lyndonne MacLeod and Carolyn's secretary June Blackburn, for their help and support during the years.

Thomas M. Connolly
Carolyn E. Begg
Glasgow, April 2001

Publisher's Acknowledgements

The publishers are grateful to the following for permission to reproduce copyright material:
The McGraw-Hill Companies, Inc., New York for Figure 18.9, reproduced from BYTE Magazine, June 1997. Reproduced with permission. © by The McGraw-Hill Companies, Inc., New York, NY USA. All rights reserved; certain portions of copyrighted Oracle Corporation user documentation or screen displays of software programs have been reproduced herein with the permission of Oracle Corporation; screenshots reprinted with permission from Microsoft Corporation.

Trademark Notice

The following are trademarks or registered trademarks of their respective companies:
ActivePerl is a trademark of ActiveState Corporation; ActiveX, Authenticode, BizTalk, FoxPro, FrontPage, Microsoft, MS-DOS, Visual Basic, Visual Studio, Windows and Windows NT are trademarks of Microsoft Corporation; ADABAS is a trademark of Software AG; Aix, CICS, DB2, DRDA, Encina, IBM, IMS, Poet, SQL/DS, systems Applications Architecture, TXSeries and WebSphere are trademarks of International Business Machines Corporation; AltaVista and Forte are trademarks of Digital Equipment Corporation; AppleTalk and LocalTalk are trademarks of Apple Computer, Inc.; Archie is a trademark of Bunyip Information Services; DBase IV is a trademark of Borland International, Inc.; DECNet is a trademark of Compaq; Delphi, InterBase and VisiBroker are trademarks of Borland Software Corporation; Dreamweaver is a trademark of Macromedia, Inc.; Enterprise JavaBeans, Java, Java Blend and JDBC are trademarks of Sun Microsystems, Inc.; Excite is a trademark of Excite, Inc.; GemStone is a trademark of Gemstone Systems, Inc.; HP Intelligent Warehouse is a trademark of Hewlett-Packard Company; IDMS, IDMS/R and IDMS/SQL are trademarks of Cullinet Corporation; Informix is a trademark of Informix Corporation; INGRES is a trademark of Computer Associates International, Inc.; Intersolv is a trademark of Merant Solutions, Inc.; ITASCA is a trademark of IBEX Knowledge Systems SA; JRun is a trademark of Allaire Corporation; Linux is a trademark of Linus Torvalds; LiveWire Pro and Netscape are trademarks of Netscape Communications Corporation; Lycos is a trademark of Lycos, Inc.; Model 204 is a trademark of Computer Corporation of America; O_2 is a trademark of O_2 Technology; Objectivity/DB is a trademark of Objectivity, Inc.; ObjectStore is a trademark of eXcelon Corporation; ObjectPAL is a trademark of Object Design; ONTOS is a trademark of Ontologic, Inc.; ORACLE and Object SQL are trademarks of Oracle Corporation; Paradox is a trademark of Corel Corporation; PowerBuilder is a trademark of Powersoft, Inc.; R:base is a trademark of R:BASE Technologies; ServletExec is a trademark of New Atlanta Communications, LLC; Simula is a trademark of Simula AS; Smalltalk is a trademark of Xerox Corporation; Sybase is a trademark of Sybase, Inc.; Tandem is a trademark of Tandem Computers, Inc.; TOPLink is a trademark of The Object People, Inc.; Tuxedo is a trademark of BEA Systems, Inc.; UniSQL/M is a trademark of UniSQL, Inc.; Unix is a trademark of the Open Group; Versant is a trademark of Versant Object Technologies; Yahoo is a trademark of Yahoo, Inc.