数 据 库

——原理、编程与性能

（第二版 影印版）

DATABASE

Principles, Programming, and Performance

(Second Edition)

Patrick O'Neil

Elizabeth O'Neil

高等教育出版社
Higher Education Press

MK Morgan Kaufmann Publishers

数 据 库

——原理、编程与性能

（第二版 影印版）

DATABASE

Principles, Programming, and Performance

(Second Edition)

图字：01-2001-1038 号

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

前　　言

　　20世纪末，以计算机和通信技术为代表的信息科学和技术，对世界的经济、军事、科技、教育、文化、卫生等方面的发展产生了深刻的影响，由此而兴起的信息产业已经成为世界经济发展的支柱。进入21世纪，各国为了加快本国的信息产业，加大了资金投入和政策扶持。

　　为了加快我国信息产业的进程，在我国《国民经济和社会发展第十个五年计划纲要》中，明确提出"以信息化带动工业化，发挥后发优势，实现社会生产力的跨越式发展。"信息产业的国际竞争将日趋激烈。在我国加入WTO后，我国信息产业将面临国外竞争对手的严峻挑战。竞争成败最终将取决于信息科学和技术人才的多少与优劣。

　　在20世纪末，我国信息产业虽然得到迅猛发展，但与国际先进国家相比，差距还很大。为了赶上并超过国际先进水平，我国必须加快信息技术人才的培养，特别要培养一大批具有国际竞争能力的高水平的信息技术人才，促进我国信息产业和国家信息化水平的全面提高。为此，教育部高等教育司根据教育部吕福源副部长的意见，在长期重视推动高等学校信息科学和技术的教学的基础上，将实施超前发展战略，采取一些重要举措，加快推动高等学校的信息科学和技术等相关专业的教学工作。在大力宣传、推荐我国专家编著的面向21世纪和"九五"重点的信息科学和技术课程教材的基础上，在有条件的高等学校的某些信息科学和技术课程中推动使用国外优秀教材的影印版进行英语或双语教学，以缩短我国在计算机教学上与国际先进水平的差距，同时也有助于强化我国大学生的英语水平。

　　为了达到上述目的，在分析一些出版社已影印相关教材，一些学校已试用影印教材进行教学的基础上，教育部高等教育司组织并委托高等教育出版社开展国外优秀信息科学和技术优秀教材及其教学辅助材料的引进研究与影印出版的试点工作。为推动用影印版教材进行教学创造条件。

　　本次引进的系列教材的影印出版工作，是在对我国高校的信息科学和技术专业的课程与美国高校的进行对比分析的基础上展开的；所影印出版的教材均由我国主要高

校的信息科学和技术专家组成的专家组，从国外近两年出版的大量最新教材中精心筛选评审通过的内容新、有影响的优秀教材；影印教材的定价原则上应与我国大学教材价格相当。

教育部高等教育司将此影印系列教材推荐给高等学校，希望有关教师选用，使用后有什么意见和建议请及时反馈。也希望有条件的出版社，根据影印教材的要求，积极参加此项工作，以便引进更多、更新、更好的外国教材和教学辅助材料。

同时，感谢国外有关出版公司对此项引进工作的配合，欢迎更多的国外公司关心并参与此项工作。

教育部高等教育司
二〇〇一年四月

Foreword

by Jim Gray, *Microsoft Research*
Series Editor, Morgan Kaufmann Series in Data Management Systems

S ince its first appearance in 1994, the O'Neil database book has become a standard text and reference for anyone learning, designing, or managing relational database applications. The book carefully presents both the theory and practice of database design and implementation. It covers relational theory, database design, database implementation, and performance-tuning issues. In all cases it starts with the general concept and then translates the ideas into specific examples demonstrated by real systems.

This revised edition reflects the substantial progress and ferment in the database field in the six years since the first edition. It places major emphasis on the object-relational model; it presents many new concepts now common in systems like Oracle, DB2, and Informix; it updates the presentation of isolation techniques; and it modernizes the presentation of performance issues. The object-relational presentation is especially noteworthy: this is widely considered the most important change to the SQL database language since the original standard. The book presents the SQL-99 design and relates that design to the principle ideas and to the major commercial products.

Professors Pat and Elizabeth O'Neil have an unusually broad and deep view of the database design issues and have been active contributors to the database field for three decades. They have taught generations of students, they have written many of the seminal research papers in the field, they have worked on the development of several products, and they have consulted with most of the vendors. They continue to innovate to this day—this book is an attempt to provide a unified view of the many disparate ideas and trends in the database area. The revision gives a modern view of the scene.

Database: Principles, Programming, and Performance makes an excellent text for anyone just approaching database systems. It is an accessible refresher for those of us who have not been paying careful attention to developments in this area and is a useful reference for designers and implementors who need just-in-time education.

Preface

Of making many books, there is no end; and much study is a weariness of the flesh.
—Christian Bible, Ecclesiastes 12:12.

*Before I came here I was confused about this subject. Having listened to your
lecture, I am still confused, but on a higher level.*
—Enrico Fermi

The goal of this text in its first edition was to introduce the reader to the fundamental principles of database theory, together with an understanding of the connections (and gaps) between theory and commercial practice. The authors believe that the high pace of change in the various commercial database system products over the last five years has made it all the more important to provide readers with an up-to-date understanding of current database practice. While a number of important changes in the database field have driven the development of this second edition, we have tried not to lose sight of the fundamental aims of this text.

As in the first edition, we provide an up-to-date introduction to SQL and to practical applications created in real commercial database systems. Whether you are an Interactive SQL user, an application programmer, a database administrator (DBA), or a student interested in learning about the field, this book was written with you in mind. The combination of practical information with underlying principles has been extended, providing what past readers have considered to be the fundamental strength of this book.

Practitioners need an introduction to database application programming that is more than the simple listing of Embedded SQL features so often seen in texts. Vendor

manuals sometimes give an excellent introduction to SQL and database programming but lack the intellectual grounding in fundamental principles that students and professionals need to adapt to future changes in database systems and languages. Some principles that are not well covered in many introductory database texts or manuals are these: implication of deadlock aborts (the need to retry), entity-relationship modeling and normalization (as well as the translation of database design into actual tables), problems of user interaction during transactions that access popular data (interaction while a transaction is active is dangerous), and finally, considerations of indexing and query optimization in terms of how they affect query performance, an important consideration for database administrators.

This book can be thought of as a solid introduction to the skills needed by a database administrator, application programmer, or sophisticated SQL user. Although the needs of a database administrator may be broader than those of an application programmer, programmers are more effective when they have a grasp of the general concepts that a DBA finds essential. The same can be said for serious Interactive SQL users. In addition to a firm grounding in SQL, an understanding of logical database design, physical layout of data, indexing, security, and cost-performance will benefit anyone working with a database system.

The authors are professors of computer science, with extensive experience consulting with database companies and database-intensive applications. See our home pages, www.cs.umb.edu/~poneil and ~eoneil.
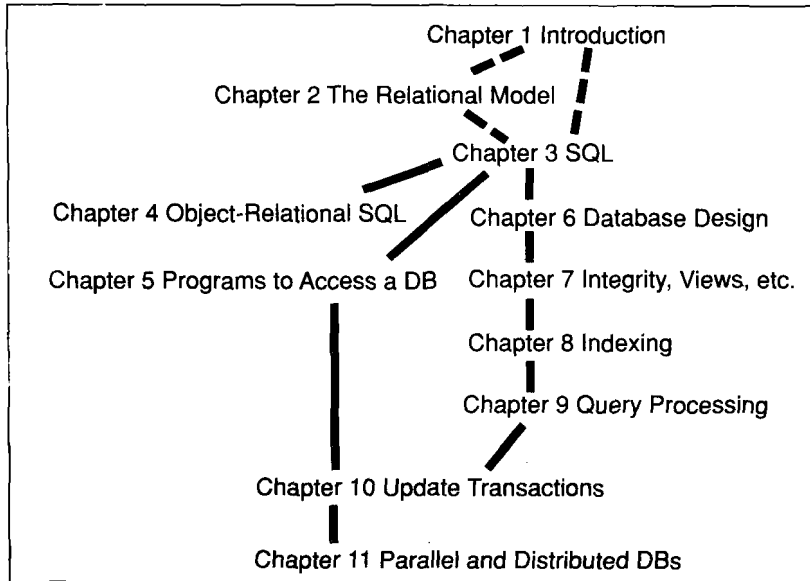
Use of This Book

This book was developed over a period of several years for an introductory database course and a subsequent, more advanced course offered at the University of Massachusetts at Boston. The first course is an introduction to database principles. Roughly the first six chapters of the text contain the material from that course. The second course deals with more advanced database concepts, with a focus on cost-performance concerns. This material is covered in Chapters 7 through 10.

It is not necessary to proceed through the book sequentially, and the chapters have been prepared to accommodate a variety of reader interests and course plans. Depending on your experience and objectives, chapters can be read briefly or in a sequence different from the order offered by the book. For example, readers who have a familiarity with basic concepts may want to begin with their specific interest in the later chapters, referring to Chapters 1 through 6 only when needed (see Figure 1 for chapter dependencies). The book has been written so that new concepts are only introduced when previous concepts have been mastered. For this reason, experienced readers can begin with the subject most appropriate to their needs.

We've intended this book either as a tutorial and reference for the professional, or as a one- or two-term introductory course in colleges and universities. The presentation moves from basic theory and concepts to the most recent developments in the field. The text introduces basic SQL simultaneously with the fundamentals of relational databases.

Chapter 1 Introduction

Chapter 2 The Relational Model

Chapter 3 SQL

Chapter 4 Object-Relational SQL   Chapter 6 Database Design

Chapter 5 Programs to Access a DB   Chapter 7 Integrity, Views, etc.

Chapter 8 Indexing

Chapter 9 Query Processing

Chapter 10 Update Transactions

Chapter 11 Parallel and Distributed DBs

Figure 1   Chapter Dependencies for Planning Reading Order

Examples from **ORACLE, INFORMIX, DB2,** and others are used to illustrate the concepts and to clarify the cost-performance issues by comparing the differing approaches used in these successful systems. The key issues in each chapter are reinforced by programming examples and exercises. Three appendices provide additional background. As an aid to self-study, solutions to selected exercises are included in the back of the book.

Discussion of New Coverage: Object-Relational Model

Since the first edition was published, the SQL language has added a number of new features, and these are covered in new sections of Chapter 3. More fundamentally, the object-relational model has become the new database standard; and while all new ORDBMS products are backward compatible so that the relational model continues to be supported, it is our contention that this change is in the process of revolutionizing the industry. Table design and SQL access to the data can be expected to develop in new and relatively unexpected ways.

The history of how the object-relational model has entered the mainstream is instructive. In 1996, **INFORMIX** Corporation purchased the Illustra product and has been working since then to consolidate object-relational features into **INFORMIX.** (Illustra, then called Montage, was covered in Section 3.11 of our first edition, but we have modified and greatly extended this material to create a new Chapter 4 for the current edition.) In 1997, **ORACLE** Corporation shipped **ORACLE** release 8, which supported a set of new object-relational features that was considered revolutionary by many. With this event,

the conversion from the relational to the object-relational model in the database industry was confirmed. Most recently, IBM's **DB2 UDB**, which has had an excellent programming interface for object-relational programming for some time, has begun to incorporate the object-relational data model at the design and interactive layers. (See [2] in the "Suggestions for Further Reading" at the end of this preface.) Unfortunately, all of these products are quite different in syntax, and a portable application is still impossible: no meaningful standardization of object-relational SQL has occurred as yet. Thankfully, the ANSI SQL-3 effort, after an extremely long process of development, released a final version of its object-relational standard in 1999. This new standard is known as SQL-99 (see [3]).

There is a tremendous pedagogical challenge to introducing new students to a field that is fragmented between two distinct models of data representation. At the present time (2000), most commercial database applications use the pure relational model. This means that object-relational concepts will be confusing to many users unless the introduction of these concepts is carefully segregated from relational concepts. We have decided in our second edition to keep our first edition introduction to the relational model in Chapter 2, and to merely bring up to date the relational SQL presentation in Chapter 3. We then expand the object-relational coverage of Section 3.11 of our first edition to a rather long Chapter 4, where we introduce **ORACLE** and **INFORMIX** object-relational conventions. Because the two products are so different, we break each section of Chapter 4 into two parts, to introduce **ORACLE** and **INFORMIX** concepts in parallel. We do not give **DB2 UDB** object-relational syntax equal weight in the current edition because the **DB2** object-relational model was still in development at the time we wrote Chapter 4.

After a thorough introduction to object-relational concepts and product usage in Chapter 4, the following chapters remain basically independent of these concerns. There are a number of reasons for this. Some of the topics covered in later chapters, such as logical database design in Chapter 6, are not well understood for the object-relational model at the present time, and for other coverage, such as update transactions in Chapter 10, object-relational concepts (aside from the introduction to PL/SQL and SPL in Chapter 4) seem largely irrelevant. A practitioner who reads through Chapter 4 carefully will be able to use object-relational SQL to augment most of the coverage of the following chapters, and for those readers who do not yet deal with object-relational products, we didn't want to confuse later issues with terminology they might wish to avoid for now. Thus we see in Figure 1 that there is no follow-on dependency on Chapter 4: the concepts of that chapter stand alone. (Chapter 7 does have coverage of catalog tables for object-relational schema objects, but this can be skipped if desired.) It is likely that the next edition will deal with object-relational concepts in all chapters of the text because it will likely be used nearly universally, but we feel that the time for that has not yet arrived.

To write the second edition, a number of difficult decisions had to be made, the most difficult being the selection of the database products and product categories to be covered in this text. We have avoided OODBMS coverage because we had to leave something out, and the potential commercial growth in the OODBMS field, which has been heralded for so many years, has never materialized; OODBMS products remain a niche category. We also decided reluctantly against providing an introduction to the Microsoft

SQL Server product, in spite of the great interest it has engendered. We made this decision because our major thrust in the second edition has been to introduce the new object-relational product features, and SQL Server doesn't yet support these features. We also decided to discontinue coverage of **INGRES**, which was central to our first edition because it was the most common database product in college use. Many other popular database products are now offering inexpensive versions for the college market, and we have therefore dropped **INGRES** in the second edition.

A Chapter-by-Chapter Outline of Changes

We outline some of the new material that has been included for the first time in the current edition in the following chapter descriptions.

Chapter 1   Introduction. New discussion of the object-relational model.

Chapter 2   The Relational Model. Aside from clarifying details from the first edition, this chapter is essentially unchanged, in keeping with our decision to cover the relational model in Chapters 2 and 3 and putting off object-relational features until Chapter 4.

Chapter 3   Basic Query Language SQL. There are a number of new features added to SQL that are covered for the first time in the second edition. Section 3.6 introduces some "Advanced SQL Syntax" not supported by all database products. This includes the INTERSECT [ALL] and EXCEPT [ALL] operations, and the new *tableref* definition of Figure 3.11, which provides for new Join Forms such as [INNER |{LEFT | RIGHT | FULL} [OUTER]] Joins, and joins ON search conditions. Since no SQL standard has been uniformly adopted by the vendors, the second edition provides a general form we call "Basic SQL," which provides the set of features we found commonly adopted by all database products under consideration.

Chapter 4   Object-Relational SQL. This is an entirely new chapter, and we provide here an extended table of contents of the sections and subsections of the chapter. Once features of **ORACLE** and **INFORMIX** are covered, a side-by-side comparison of features ends each section.

    4.1   Introduction: Definitions and Object-Relational History

    4.2   Object types in **ORACLE**; row types in **INFORMIX**; use of object (row) types for table definitions; object nesting; dot notation to access columns; lack of encapsulation of object data; coverage of REFs in **ORACLE** but not in **INFORMIX**; type inheritance in **INFORMIX** but not in **ORACLE**.

4.3 Collection Types. **ORACLE** has two collection types: nested tables (a column value can be a table) and VARRAYs (array type column value). **INFORMIX** has three collection types: sets, multisets (unordered, like sets, with duplicates allowed), and lists (ordered). Both allow ad hoc queries to retrieve data from collections, and to insert and update collections.

4.4 User-Defined Functions (UDFs) and Methods. **ORACLE** has a procedural SQL language PL/SQL and **INFORMIX** has one called SPL. In both products, UDFs can be written in this procedural language (or an Embedded SQL language such as Java), which can be called like built-in functions within SQL. Methods are UDFs supported in **ORACLE** that are defined as part of an object type.

4.5 External Functions and Packaged User Defined Types (UDTs). We survey the database systems' capabilities of "packaging" user-defined types with a set of external functions, which are UDFs written in a language like C and made available to the database server. Such a package is called a Cartridge in **ORACLE**, a DataBlade in **INFORMIX**, and an Extender in **DB2 UDB**.

The following chapter numbers increment by one the chapter numbers of the first edition because of the addition of the new Chapter 4.

Chapter 5   Programs to Access a Database. More programming examples have been provided, in particular, on transactions. We have also improved the coverage of error handling. **ORACLE** syntax has been brought up to date, in particular the **ORACLE** Dynamic SQL syntax, including SQLSTATE. **DB2 UDB** is also covered now (replacing **INGRES** in the first edition).

Chapter 6   Database Design. This is logical database design, including the E-R model and normalization. A number of definitions and proofs have been clarified and illustrations added.

Chapter 7   Integrity, Views, Security, and Catalogs. A number of standard clauses of the Create Table and Alter Table statements have been brought up to date. The section on triggers has been rewritten with real examples from **ORACLE** and **DB2 UDB**. The restrictions on updatable views have been updated significantly, and there are no longer any restrictions on querying views. The section on system catalogs has been slightly expanded and now includes a subsection on object-relational catalog coverage.

Chapter 8   Indexing. Figures for disk access speed, disk capacity, disk cost, and memory cost have been updated. The **ORACLE** Create Tablespace statement has been

updated. All specialized **INGRES** indexing capabilities (ISAM, hashing, etc.) have been dropped, and **ORACLE** index-organized tables and table clusters, in particular hash clusters, are now covered. The discussion of overflow chaining has been modified to deal with the new. **ORACLE** hash clusters architecture. **DB2 UDB** index structures are now covered.

Chapter 9   Query Processing. While the earlier parts of this chapter have been brought up to date to deal with newer product features, at the insistence of a number of reviewers, the coverage of the IBM mainframe **DB2** query features leading up to Set Query benchmark results from the first edition have been retained. Because most of these **DB2** query capabilities are still state-of-the-art, and performance is explained at a level of detail that would be hard to duplicate today, this material can still be used to teach students important concepts about how queries perform. In fact, the coverage of the query processor is still relevant to the current mainframe **DB2** product, **DB2 for OS/390.**

Chapter 10 A number of improvements have been made in definitions and proofs of this chapter. In Section 10.5, the Levels of Isolation definitions have been modified to take into account newly discovered facts in [1].

Chapter 11 Very minor changes have been made.

Support on the World Wide Web

The home page for this book is http://www.mkp.com/books_catalog/1-55860-438-3.asp. This page will link to the authors' Web site and will provide database create and load scripts for several products, current errata for the text, example programs, and slides to use in lectures. Instructors will also be able to obtain solutions to non-dotted exercises. Electronic mail on suggestions, contributions of teaching materials, or errata should be addressed to poneil@cs.umb.edu or eoneil@cs.umb.edu.

Acknowledgments

It is our pleasure to acknowledge the help of many people who gave suggestions and help in producing this book. A number of students helped by pointing out errors in the first edition text and notes for the second edition. The most prominent among these are Christian Junghanss, Dimitrios Liarokapis, and Usha Rao. Several colleagues also provided valuable help for the second edition, including Michael Carey, Don Chamberlin, Andrew Eisenberg, Jim Gray, Berl Hartman, Rick Martin, Jim Melton, Betty Salzberg, Mike Ubell, and Gerhard Weikum, as well as numerous reviewers that read over drafts of the manuscripts for Morgan Kaufmann. We received help during our work from **IBM**, **INFORMIX**, and **ORACLE**. We are also grateful for past help from Henry Etlinger, Goetz

Graefe, Fred Korz, Edward Omiecinski, Eugene O'Neil, Julie Pabst, Bryan Pendleton, Donald Slutz, Bruce Spatz, David Spooner, Toby Teorey, Gottfried Vossen, and Yun Wang.

Patrick O'Neil and Elizabeth O'Neil
University of Massachusetts at Boston
poneil@cs.umb.edu, eoneil@cs.umb.edu

Suggestions for Further Reading

[1] Hal Berenson, Phil Bernstein, Jim Gray, Jim Melton, Elizabeth O'Neil, and Patrick O'Neil. "A Critique of ANSI SQL Isolation Levels." *ACM SIGMOD Proceedings,* May 1995, pp. 1–10.

[2] Michael Carey, Don Chamberlin, et al. "O-O, What Have They Done to DB2?" *Proceedings of the 1999 VLDB Conference.*

[3] Andrew Eisenberg and Jim Melton. "SQL:1999, formerly known as SQL3." *SIGMOD Record,* vol. 28, no. 1, March 1999, pp. 131–138.

Contents