THOMAS CONNOLLY ▸ CAROLYN BEGG ▸ ANNE STRACHAN

DATABASE SYSTEMS

A Practical Approach to Design, Implementation and Management

ADDISON-WESLEY

Database Systems

A Practical Approach to Design, Implementation and Management

Thomas M. Connolly

Carolyn E. Begg

Anne D. Strachan

University of Paisley

ADDISON-WESLEY PUBLISHING COMPANY
Wokingham, England • Reading, Massachusetts • Menlo Park, California
New York • Don Mills, Ontario • Amsterdam • Bonn • Sydney • Singapore
Tokyo • Madrid • San Juan • Milan • Paris • Mexico City • Seoul • Taipei

Cover designed by op den Brouw, Design & Illustration, Reading
and printed by The Riverside Printing Co. (Reading) Ltd
Typeset by Colset Pte Ltd, Singapore
Printed and bound by Butler & Tanner, Frome, Somerset

First printed 1995

ISBN 0-201-42277-8

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

Library of Congress Cataloging-in-Publication Data applied for

Publisher's acknowledgements
The publisher wishes to thank the following for permission to reproduce the material. Figure 4.6 material from: Earl, M.J. (1989). *Management Strategies for Information Technology*. Hemel Hempstead: Prentice-Hall International (UK) Ltd; Loring, P. and De Garis, C. (1992). 'The changing face of Data Administration' in Clarke, R. and Cameron, J. (eds) *Managing Information Technology's Organisational Impact II*, **A** (3), J.F.I.P.: Amsterdam: Elsevier Science.

Database Systems

INTERNATIONAL COMPUTER SCIENCE SERIES

Consulting Editor **A D McGettrick** University of Strathclyde

SELECTED TITLES IN THE SERIES

Software Development with Z *J Wordsworth*

Program Verification *N Francez*

Performance Modelling of Communication Networks *P Harrison and N Patel*

Concurrent Systems: An Integrated Approach to Operating Systems, Database, and Distributed Systems *J Bacon*

Introduction to Parallel Processing *B Codenotti and M Leoncini*

Concurrent Programming *A Burns and G Davies*

Comparative Programming Languages (2nd edn) *L Wilson and R Clark*

Functional Programming *R Plasmeijer and M van Eekelen*

Object-Oriented Database Systems: Concepts and Architectures *E Bertino and L D Martino*

Programming in Ada (4th edn) *J Barnes*

Software Design *D Budgen*

Ada from the Beginning (2nd edn) *J Skansholm*

Programming Language Essentials *H E Bal and D Grune*

Human-Computer Interaction *J Preece et al.*

Distributed Systems: Concepts and Design (2nd edn) *G Coulouris, J Dollimore and T Kindberg*

Fortran 90 Programming *T M R Ellis, I Philips and T Lahey*

Parallel Processing: The Transputer and its Applications *E Hull, D Crookes and P Sweeney*

Foundations of Computing: System Development with Set Theory and Logic *T Scheurer*

To Sheena, for her patience and understanding during the last three years.
To Kathryn, for the constant pleasure she has given us since her birth.
To my Mother, who died during the writing of this book - sleep peacefully, Mum.
Thomas M. Connolly

To my Mother and Father and, in particular, to Alan, for his support and understanding.
Carolyn Begg

To Paul, for his encouragement and support.
Anne Strachan

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 these 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 objective of this book, therefore, is to provide a textbook that introduces the theory behind databases as clearly as possible and, in particular, provides 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 two main phases: logical database design 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 into a logical data model by removing constructs that cannot be represented in relational systems. In the second 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 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 checklist. To help the reader understand these important issues, the book has two chapters providing comprehensive worked examples, based on an integrated case study, *DreamHome*. In addition, a second case study, *Wellmeadows Hospital*, is provided to allow readers to try out the methodology for themselves.

The book also examines in some depth:

- the latest standard in SQL, SQL-92, including embedded SQL;
- the concepts and problems with the emerging distributed database system and object database system;
- the new Object Database Management Group standard for object database systems.

Intended Audience

This book is intended to be used as a textbook for a one- or two-semester course in database management or database design in an introductory undergraduate course, in a 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 A before covering the material in Chapter 9 on physical database design. This background ideally will have been obtained from a prior course. If this is not possible, then the material in Appendix A 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 Sections 12.5 and 12.6 on embedded and dynamic SQL.

Distinguishing Features

(1) An easy-to-use, step-by-step methodology for logical database design, based on the widely-accepted Entity–Relationship model, with normal-

ization used as a validation technique. There is an accompanying chapter showing the methodology in use and a separate chapter showing how database design fits into the overall systems development lifecycle.

(2) An easy-to-use, step-by-step methodology for physical database design, covering the mapping of the logical design to a physical implementation, selecting file organizations and indexes appropriate for the applications, and when to introduce controlled redundancy. There is an accompanying chapter showing the methodology in use.

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

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

(5) Two tutorial-style chapters on the new SQL (SQL–92) standard, covering both interactive and embedded SQL.

(6) A tutorial-style chapter on QBE.

(7) Comprehensive coverage of the relational data model.

(8) Comprehensive coverage of the concepts and issues relating to the increasingly important area of distributed database management systems.

(9) Comprehensive introduction to the concepts and issues relating to the increasingly important area of object-oriented database systems, including a review of the ODMG standard and a preview of the next SQL standard, SQL3.

(10) Introduction to DBMS system implementation concepts, including concurrency and recovery control, and security and integrity.

(11) An overview of legacy systems and a detailed comparison of the three traditional data models.

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, whatever their background and experience, to follow and understand. From the authors' experience of using textbooks, which clearly 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. The style used is that such concepts are boxed.
- Diagrams are liberally used throughout to help support and clarify concepts.
- A very practical orientation. To this end, each chapter contains many worked examples to help illustrate the concepts covered.
- A summary at the end 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 individuals to demonstrate and test the individual's understanding of the chapter.

Instructor's Manual

A comprehensive supplement containing numerous instructional resources is available for this textbook, upon request from Addison-Wesley. The accompanying manual includes:

- *Course structures* This includes 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 problems (similar to the questions and exercises from the text) with solutions.
- *Transparency masters* A set of masters for overhead transparencies of enlarged illustrations and tables from the text help the instructor to associate lectures and class discussion to material in the text. There is also a set of transparencies containing lecture notes for the main chapters in this book.

Organization of this Book

Part 1 Background

Part 1 of the book serves to introduce the field of database systems and database design, introducing the relational model, which is the main focus of attention.

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. It provides a description of *DreamHome*, a case study that is used extensively throughout the book. It also provides a second case study, the *Wellmeadows Hospital*, which can be used as a student project.

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.

Chapter 3 introduces the concepts behind the relational model, the most popular data model at present, and the one most 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.

A short overview of relational algebra and relational calculus is presented with examples to illustrate all the operations. This could be omitted for a first course in database management. However, relational algebra is needed to understand fragmentation in Chapter 16 on distributed databases. 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 11 and 12, although not essential. The chapter concludes with an overview on views, which is expanded upon in Chapter 12.

Chapter 4 completes the introduction to the first part of the book. This chapter presents an overview of the main phases of the information systems lifecycle, and discusses how these relate to the development of database applications. In particular, it emphasizes the importance of database design and shows how the process can be decomposed into two stages: logical database design and physical database design. It also shows how the design of the application (the functional approach) affects database design (the data approach). The chapter also examines the applicability of Computer-Aided Software Engineering (CASE) tools to the database lifecycle.

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 discussion of the Data Administrator (DA) and the Database Administrator (DBA), the main personnel responsible for the planning, design and administration of a database.

Part 2 Methodology

Part 2 of the book presents a methodology for both logical and physical database design for relational systems.

Chapter 5 covers the concepts of Chen's Entity-Relationship (ER) model, and the Enhanced Entity-Relationship (EER) model, which allows more advanced data modelling using subclasses and superclasses. The EER model is a popular high-level conceptual data model and is a fundamental technique of the logical database design methodology presented herein. A

worked example taken from the *DreamHome* case study is used to demonstrate how to create an EER model.

Chapter 6 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 each of the normal forms up to, and including, Boyce–Codd normal form.

Chapter 7 presents a step-by-step methodology for logical database design for the relational model. 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. It shows how to create a data model for each user view, validate it, and then map it to a data model suitable for implementation in a relational system. To complete the logical design methodology, the chapter shows how to merge the resulting data models together into a global data model that represents the part of the enterprise being modelled.

Chapter 8 provides a realistic worked example of the logical database design methodology taken from the *DreamHome* case study. It illustrates the creation and validation of local data models for two user views, and illustrates how to merge the resulting views together.

Chapter 9 presents a step-by-step methodology for physical database design and implementation for relational systems. It shows how to take the global data model developed during logical database design and how, using a variety of techniques, to design and implement it in 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.

Chapter 10 provides a realistic worked example of the physical database design methodology taken from the *DreamHome* case study. It illustrates the implementation of part of the global logical data model derived in Chapter 8 using the PC database management system, Paradox for Windows.

Part 3 Database Languages

Part 3 of the book looks at the two main languages of relational systems: SQL and QBE.

Chapter 11 introduces the new 1992 SQL standard, SQL-92. The chapter is presented as a worked tutorial, giving a series of worked examples that demonstrate the main concepts of SQL. In particular, it concentrates on the data manipulation statements: SELECT, INSERT, UPDATE and DELETE. It also covers the SQL-92 data types and shows basic forms of the data definition statements.

Chapter 12 covers the more advanced features of the SQL-92 standard. Again, the chapter is presented as a worked tutorial. This chapter looks at views, the Integrity Enhancement Feature (IEF) and the more advanced features of the data definition statements, including the access control state-

ments GRANT and REVOKE. There are two sections that examine embedded and dynamic SQL, with sample programs in 'C'. For an introductory course in databases, these two sections could be omitted.

Chapter 13 is another practical chapter that looks at 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 demonstrated using Paradox for Windows 5.1.

Part 4 Selected Database Issues

Part 4 of the book examines four specific topics that the authors consider necessary for a modern course in database management.

Chapter 14 is divided into two main areas: database integrity and security. Security is considered not just in the context of DBMS security but also in the context of the security of the DBMS environment. Thus, the chapter examines both computer-based and non-computer-based solutions, concluding with a presentation of risk analysis.

Chapter 15 concentrates on three of the functions that a Database Management System (DBMS) should provide, namely transaction management, concurrency control and recovery control. 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.

Distributed database management system (DDBMS) 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 16** discusses the concepts and problems of distributed database management systems, where users can not only access the database at their own site but also access data stored at remote sites. There are claims that in the next ten years centralized database systems will be an 'antique curiosity' as most organizations move towards distributed database systems.

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 database, or object-oriented database, is a major development in the database systems area that attempts to overcome these failings. **Chapter 17** examines this development in some detail. After presenting the main concepts of object orientation, the chapter examines two approaches that are being investigated: extending the relational model or producing a new data model based on object concepts. The chapter examines two new emerging but contrasting standards: the next SQL standard, called SQL3, and the new *de facto* object database standard from the Object Database Management Group (ODMG). The chapter also examines how the methodology presented in Part 2 of the book may be extended for object databases.

Appendices

Appendix A 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 9.
Appendix B introduces the basics of the network data model.
Appendix C introduces the basics of the hierarchical data model, concentrating in particular on the concepts of the hierarchical product IMS (Information Management System).
Appendix D compares and contrasts the features of the three traditional data models.
Appendix E summarizes the steps in the methodology presented in Chapter 7 and 9 for logical and physical database design.

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 Addison-Wesley, or by electronic mail to:

conn-ci0@paisley.ac.uk

Acknowledgements

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, we should like to thank Dr Simon Plumtree and Nicky Jaeger, our editors, for their help, encouragement and professionalism throughout the years. We should also like to thank the reviewers of the book, who contributed their comments, suggestions 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,

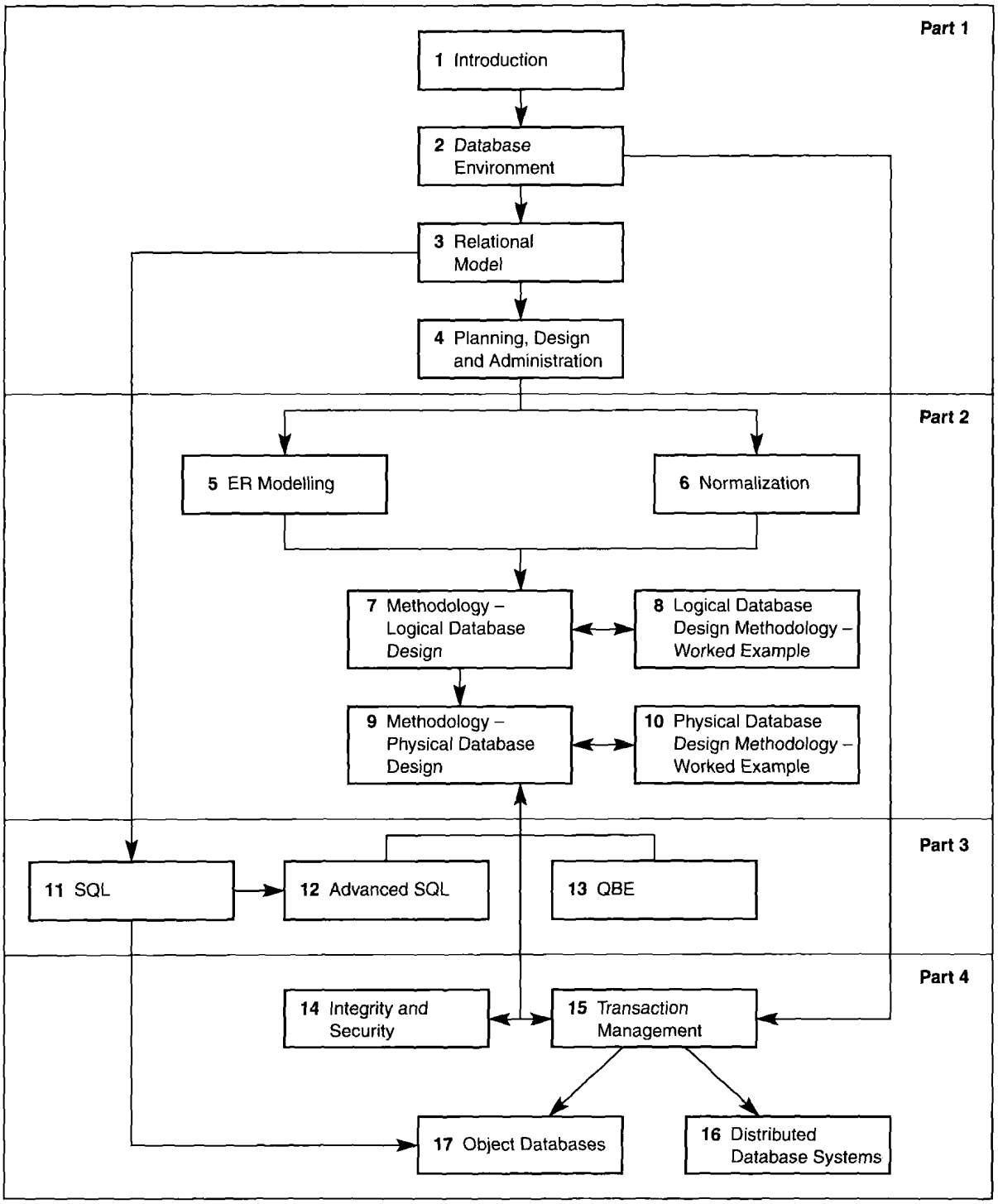

Figure P.1 Logical organization and suggested paths through it.

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. Many others are still anonymous to us - we thank you for the time you must have spent on the manuscript. We should like to express our gratitude to John Wade of IBM for reading and commenting on the IMS material. Special thanks must go to our excellent production editor, Martin Tytler, for his patience and assistance throughout the production process, and to our proofreader, Lionel Browne, for the outstanding work in spotting and resolving all the inconsistencies and irregularities in the typeset 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 Thomas's secretary Lyndonne MacLeod, for managing his administrative duties during the years.

Thomas M. Connolly
Carolyn Begg
Anne Strachan
Glasgow, September 1995

Trademark notice

Access™ and Windows™ are trademarks, and Microsoft® and MS-DOS® are registered trademarks of Microsoft Corporation
ADABAS™ is a trademark of Software AG
DB2™, IMS™, Systems Application Architecture™ and SQL/DS™ are trademarks of International Business Machines Corporation
dBase IV™ is a trademark of Ashton-Tate, Incorporated
FoxPro™ and R:base™ are trademarks of Microrim, Incorporated
GemStone™ is a trademark of Servio Logic Corporation
IDMS™, IDMS/R™ and IDMS/SQL™ are trademarks of Computer Associates International, Incorporated
IDS™ is a trademark of General Electric
Informix™ is a trademark of Informix Software, Incorporated
INGRES™ is a trademark of Ingres Corporation
ITASCA™ is a trademark of ITASCA, Incorporated
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, Incorporated
ObjectStore™ and ObjectPAL™ are trademarks of Object Design
ONTOS™ is a trademark of Ontologic, Incorporated
ORACLE® is a registered trademark of Oracle Corporation
Paradox™ is a trademark of Borland International, Incoporated
Simula™ is a trademark of Simula AS
Smalltalk™ is a trademark if Xerox Corporation
StP™ is a trademark of Interactive Development Environment
Sybase® is a registered trademark of Sybase, Incorporated
UniSQL/M™ is a trademark of UniSQL, Incorporated
UNIX™ is a trademark of AT&T Bell Laboratories
Versant™ is a trademark of Versant Object Technologies