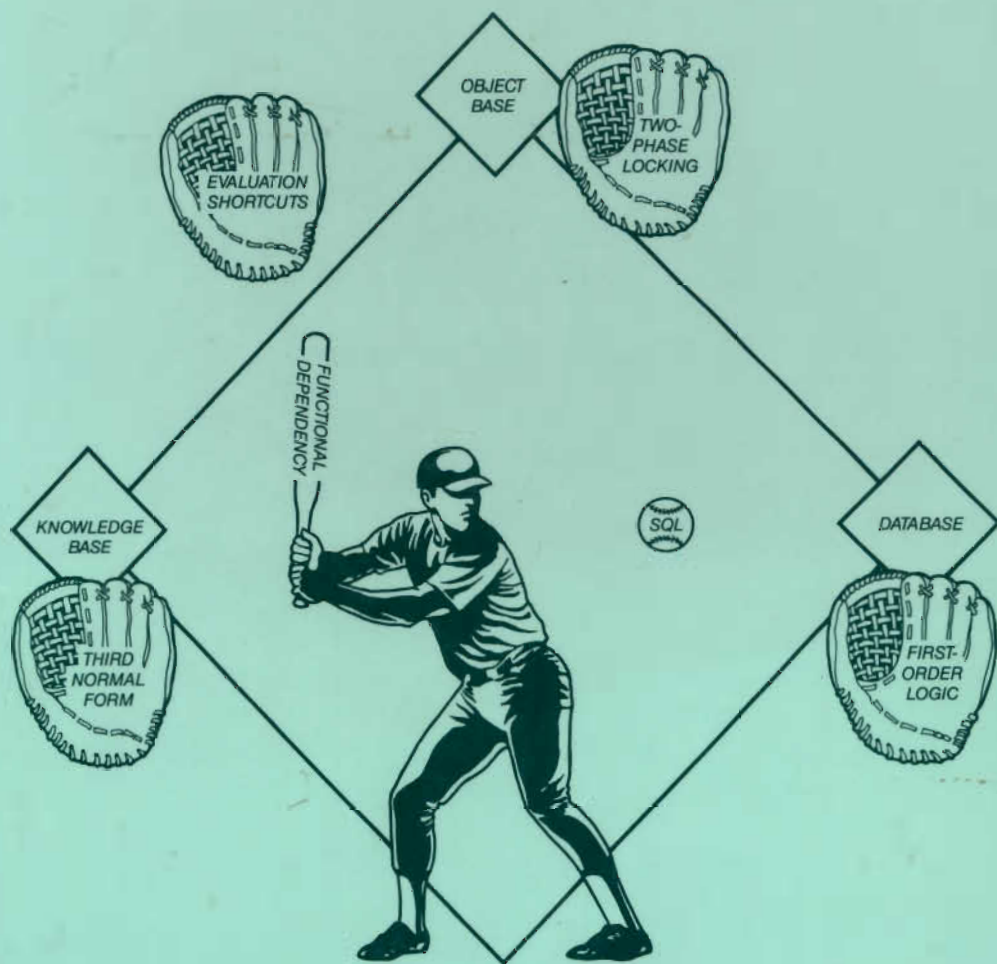


# PRINCIPLES OF DATABASE AND KNOWLEDGE - BASE SYSTEMS

VOLUME I: CLASSICAL DATABASE SYSTEMS

Jeffrey D. Ullman



COMPUTER SCIENCE PRESS

15T-2575

PRINCIPLES OF  
DATABASE AND  
KNOWLEDGE - BASE  
SYSTEMS

VOLUME I: CLASSICAL DATABASE SYSTEMS

---

Jeffrey D. Ullman  
*STANFORD UNIVERSITY*

COMPUTER SCIENCE PRESS

# TABLE OF CONTENTS

## **Chapter 1: Databases, Object Bases, and Knowledge Bases 1**

- 1.1: The Capabilities of a DBMS 2
- 1.2: Basic Database System Terminology 7
- 1.3: Database Languages 12
- 1.4: Modern Database System Applications 18
- 1.5: Object-base Systems 21
- 1.6: Knowledge-base Systems 23
- 1.7: History and Perspective 28
  - Bibliographic Notes 29

## **Chapter 2: Data Models for Database Systems 32**

- 2.1: Data Models 32
- 2.2: The Entity-relationship Model 34
- 2.3: The Relational Data Model 43
- 2.4: Operations in the Relational Data Model 53
- 2.5: The Network Data Model 65
- 2.6: The Hierarchical Data Model 72
- 2.7: An Object-Oriented Model 82
  - Exercises 87
  - Bibliographic Notes 94

## **Chapter 3: Logic as a Data Model 96**

- 3.1: The Meaning of Logical Rules 96
- 3.2: The Datalog Data Model 100
- 3.3: Evaluating Nonrecursive Rules 106
- 3.4: Computing the Meaning of Recursive Rules 115
- 3.5: Incremental Evaluation of Least Fixed Points 124
- 3.6: Negations in Rule Bodies 128
- 3.7: Relational Algebra and Logic 139
- 3.8: Relational Calculus 145
- 3.9: Tuple Relational Calculus 156
- 3.10: The Closed World Assumption 161
  - Exercises 164
  - Bibliographic Notes 171

<b>Chapter 4: Relational Query Languages</b>	<b>174</b>
4.1: General Remarks Regarding Query Languages	174
4.2: ISBL: A “Pure” Relational Algebra Language	177
4.3: QUEL: A Tuple Relational Calculus Language	185
4.4: Query-by-Example: A DRC Language	195
4.5: Data Definition in QBE	207
4.6: The Query Language SQL	210
4.7: Data Definition in SQL	223
4.8: Embedding SQL in a Host Language	227
Exercises	235
Bibliographic Notes	238
<b>Chapter 5: Object-Oriented Database Languages</b>	<b>240</b>
5.1: The DBTG Data Definition Language	240
5.2: The DBTG Query Language	246
5.3: The DBTG Database Modification Commands	258
5.4: Data Definition in IMS	262
5.5: A Hierarchical Data Manipulation Language	264
5.6: Data Definition in OPAL	271
5.7: Data Manipulation in OPAL	278
Exercises	288
Bibliographic Notes	292
<b>Chapter 6: Physical Data Organization</b>	<b>294</b>
6.1: The Physical Data Model	295
6.2: The Heap Organization	304
6.3: Hashed Files	306
6.4: Indexed Files	310
6.5: B-trees	321
6.6: Files with a Dense Index	328
6.7: Nested Record Structures	330
6.8: Secondary Indices	339
6.9: Data Structures in DBTG Databases	342
6.10: Data Structures for Hierarchies	346
6.11: Data Structures for Relations	351
6.12: Range Queries and Partial-match Queries	354
6.13: Partitioned Hash Functions	358
6.14: A Search Tree Structure	361
Exercises	368
Bibliographic Notes	374

<b>Chapter 7: Design Theory for Relational Databases</b>	<b>376</b>
7.1: What Constitutes a Bad Database Design?	377
7.2: Functional Dependencies	379
7.3: Reasoning About Functional Dependencies	382
7.4: Lossless-Join Decomposition	392
7.5: Decompositions That Preserve Dependencies	398
7.6: Normal Forms for Relation Schemes	401
7.7: Lossless-Join Decomposition Into BCNF	403
7.8: Dependency-preserving 3NF Decompositions	409
7.9: Multivalued Dependencies	413
7.10: Fourth Normal Form	420
7.11: Generalized Dependencies	423
Exercises	435
Bibliographic Notes	441
<b>Chapter 8: Protecting the Database Against Misuse</b>	<b>446</b>
8.1: Integrity	447
8.2: Integrity Constraints in Query-by-Example	452
8.3: Security	456
8.4: Security in Query-by-Example	458
8.5: Security in SQL/RT	460
8.6: Security in OPAL/GEMSTONE	462
Exercises	464
Bibliographic Notes	466
<b>Chapter 9: Transaction Management</b>	<b>467</b>
9.1: Basic Concepts	468
9.2: A Simple Transaction Model	477
9.3: The Two-phase Locking Protocol	484
9.4: A Model with Read- and Write-Locks	486
9.5: Lock Modes	490
9.6: A Read-Only, Write-Only Model	492
9.7: Concurrency for Hierarchically Structured Items	502
9.8: Handling Transaction Failures	508
9.9: Aggressive and Conservative Protocols	511
9.10: Recovery From Crashes	516
9.11: Timestamp-based Concurrency Control	524
Exercises	535
Bibliographic Notes	540

<b>Chapter 10: Distributed Database Management</b>	<b>543</b>
10.1: Distributed Databases	543
10.2: Distributed Locking	546
10.3: Distributed Two-phase Locking	555
10.4: Distributed Commitment	557
10.5: A Nonblocking Commit Protocol	564
10.6: Timestamp-based, Distributed Concurrency	573
10.7: Recovery of Nodes	575
10.8: Distributed Deadlocks	576
Exercises	582
Bibliographic Notes	585
<b>Bibliography</b>	<b>588</b>
<b>Index</b>	<b>616</b>

## ABOUT THE BOOK

This is the first in a two-volume series intended to integrate the major trends in database systems—knowledge systems and object-oriented systems—along with classical database concepts. This volume covers data models, including the logical models upon which knowledge systems depend and object models used for object-oriented systems, as well as the classical models. Examples of commercial languages based on each of these models are included. The book also discusses physical storage techniques, design theory for relational databases, security, integrity, concurrency control, recovery in database systems, and distributed database systems.

## TO APPEAR IN VOLUME II

The second volume covers query optimization in database systems, and explains extensions of these ideas to handle the more expressive query languages that are used in knowledge-base systems. Recently discovered techniques for efficient implementation of logic languages will be discussed, along with the design of some experimental knowledge-base systems. The “universal relation” model, for understanding queries posed in natural language or in very high-level languages, will also be treated.

## ABOUT THE AUTHOR

Jeffrey D. Ullman received his B.S. degree from Columbia University in 1963 and his Ph.D. from Princeton in 1966. He was with AT&T Bell Laboratories for three years, prior to joining the faculty at Princeton University, where he taught from 1969–1979. Since 1979, he has been a professor of Computer Science at Stanford University. He currently serves on the editorial boards of *SIAM J. Computing*, *J. Computer and System Sciences*, *Theoretical Computer Science*, and *J. Parallel and Distributed Computing*, and is on the NSF advisory panel for Information, Robotics, and Intelligent Systems. In the past, Professor Ullman has served as an editor of *J. ACM*, on the examination committee for the Computer Science Graduate Record Examination, and on the Computer Science advisory panel for NSF.

Professor Ullman is coeditor, with Alfred V. Aho, of the Principles of Computer Science Series for Computer Science Press.

ISBN 0-7167-8158-1



90000 >



EAN

9 780716 781585