

Series Foreword	xvii
How to Read This Book	xix
Acknowledgment	xxi

1 AN OVERVIEW OF DATABASE MANAGEMENT

1.1	Introduction	1
1.1.1	Purpose of a Database	1
1.1.1.1	A Database Provides a Means for Answering All Questions	1
1.1.1.2	A Database Provides a Means for Horizontal Integration	2
1.1.1.3	A Database Provides a Means for Vertical Integration	3
1.1.1.4	Using a DBMS Improves the Performance of the Data Processing (DP) Department	4
1.1.1.5	A Data Dictionary Is a Benefit in Itself in Some Cases	5
1.1.2	Definition of a Database	5
1.1.3	When Is a Database Required? Subject Overview	7
1.1.3.1	Application Development Ease and Speed	7
1.1.3.2	Application Evolution Ease: Program-to-Data Independence	8
1.1.3.3	Unpredictable Queries: Completeness	8
1.1.3.4	Integrity, Backup, and Restore Capabilities (Multitasking)	9
1.1.3.5	Security	9
1.1.3.6	Improved Horizontal/Vertical Integration and Communication	10
1.1.4	Database Requirements by Application Area	10
1.1.4.1	Business DP and Performance Considerations	10
1.1.4.2	Management Information Systems (MISs)	11
1.1.4.3	Personal Databases	12
1.1.4.4	Industrial and Scientific Applications	12
1.2	Key Steps of a Database Project	13
1.2.1	Preliminary Functional Specifications	13
1.2.2	Detailed Functional Specifications	18

1.2.2.1	Contents of the Detailed Functional Specification	19
1.2.2.2	Appoint Members of a Functional Specifications Group	20
1.2.2.3	Obtain Future User Agreement	21
1.2.2.4	Comments	21
1.2.3	Documenting Acquisition Requirements for Hardware and DBMS	22
1.2.4	Choosing DBMS and Hardware	24
1.2.5	Documenting Project Plan, Costs, and Risks	25
1.2.6	Designing File Architecture	26
1.2.6.1	Derive the Conceptual Schema from the Data Dictionary	26
1.2.6.2	Derive File Access and Relationship Paths from Processing	27
1.2.6.3	Use AREA-Type Data Grouping When Available	27
1.2.6.4	Calculate DBMS Load at Peak Usage	27
1.2.7	Designing Program Architecture	27
1.2.8	Writing, Testing, and Documenting the Programs	28
1.2.9	Initial Database Loading, Parallel Running, and Testing	28
1.2.10	Writing the Documentation for Users and the DBA	29
1.2.11	Initiating the DBA Function and Starting Daily Use	29
1.3	Database Fundamentals	29
1.3.1	Views	29
1.3.1.1	Single-Level View	31
1.3.1.2	Multilevel View	31
1.3.2	The First Three Schemas of a Database	33
1.3.2.1	Data Fields and Data Groups	33
1.3.2.2	Data Existence Constraints	33
1.3.2.3	Conceptual Schema	35
1.3.2.4	Physical Schema	35
1.3.2.5	The External Schemas or Subschemas	36
1.3.3	Definition of a Relation (Static, Dynamic)	36
1.3.3.1	Relationships: 1-to-N, 1-to-1, N-to-P	39
1.3.3.2	Relational Model	40
1.3.3.3	Normal Forms of a Relation: Fourth and Other Normal Forms	41

1.3.3.4	Physical Implementation of a Relational Database	44
1.3.4	Relational Algebra	45
1.3.4.1	Defining a Relation in the Data Dictionary	45
1.3.4.2	Evolution of a Database	45
1.3.4.3	Adding, Deleting, and Modifying Tuples	45
1.3.4.4	Selecting Tuples That Satisfy a Set of Constraints	46
1.3.4.5	Projecting a Relation to Remove Unwanted Attributes	47
1.3.4.6	Joining Two Relations, Associating Matching Tuples	48
1.3.4.7	Joining Master and Event Files to Avoid the Connection Trap	50
1.3.4.8	Building the Union of Two Relations	51
1.3.4.9	Eliminating Duplicate Tuples in a Relation	51
1.3.4.10	Building the Difference of Two Relations	51
1.3.4.11	The Dynamic Model of Data	51
1.3.5	Nonrelational Models: Comparisons with Relational Models	54
1.3.5.1	The Hierarchical Model	56
1.3.5.2	The Network Model	58
1.3.5.3	The Entity-Relation Model	62
1.3.5.4	The Master-Event Model	64
1.3.5.5	Example: Education Database	65

2 DETAILED SPECIFICATION: RELATIONAL TECHNIQUES

2.1	MSD: A New Approach to Specification and Design	70
2.1.1	Requirements for the DFSs of an Interactive Application	70
2.1.1.1	DFSs Must Be Sufficiently Detailed	70
2.1.1.2	Implications of Sufficiently Detailed Specifications	72
2.1.1.3	A User-Oriented Approach: Scenarios and Dictionary	73
2.1.1.4	The Need for a Relational Model	76
2.1.2	Results of the MSD Detailed Functional Specification Phase	77

2.1.3	Database Design Phase	78
2.1.4	Program Module Design Phase	79
2.1.5	Objections to the Detailed Specification Approach of MSD	80
2.1.6	Organization Prerequisites	81
2.1.7	Scenarios	82
2.1.8	Scenario Description Rules	83
2.1.9	Sample Scenarios	86
2.2	MSD Dictionary	89
2.2.1	Computer-Supported Dictionaries	89
2.2.2	The MSD Data Dictionary	90
2.2.2.1	Identification Rules for Relations and Attributes	90
2.2.2.2	Attribute Slip	91
2.2.2.3	Relation Slip	95
2.2.3	Other MSD Descriptions	99
2.2.3.1	Screen Form and Report Layout Slips	99
2.2.3.2	Co-Routine Slips	104
2.2.3.3	Notes to the Programmer	108
2.2.3.4	Transaction Slip	109
2.2.3.5	View and Access/Relationship Path Slips	116
2.3	Conclusions on Detailed Functional Specification Methods	122
2.3.1	Information Systems	123
2.3.2	Transforming an Information System into a DP Database	125
3	DATA REPRESENTATION, PACKING, AND PROTECTION	
3.1	Data Representation and Packing	128
3.1.1	Full-Length Storage	128
3.1.2	Representing Data with Codes	128
3.1.3	Vocabulary and Alphabet	129
3.1.4	Base	130

3.1.5	Data Packing in Remote Communications	132
3.1.6	Packing Entire Records Using Multiple Bases	132
3.1.7	Packing Using Multiple Words	137
3.1.8	When Is Data Packing Worthwhile?	140
3.1.9	Reducing the Cost of Packing: Bases 2^n	142
3.1.10	Special Representation Methods for Technical Data	143
3.1.10.1	Continuous Data Streams	143
3.1.10.2	Slow-Evolution Processes	144
3.1.10.3	Coding by Exception	145
3.1.10.4	Coding by Dictionary Rank	145
3.1.10.5	Using Functions	145
3.1.10.6	Suitable Data Manipulation Technique	146
3.2	Access Security Protection	147
3.2.1	Secret Packing (Encryption)	147
3.2.2	Dynamic Password	148
3.3	Keyboard Input Protection	149
3.4	Protection of Stored and Transmitted Data	154
3.4.1	Preliminary Requirements	154
3.4.2	Design Specifications of the Verification Code (VC)	155
3.4.3	Verification Codes Form a Group	156
3.4.4	Implementation	158
3.4.5	Conclusion: Protection Quality	159
4	SELECTION IN A DATABASE: A COMPLETE DISCUSSION	
4.1	Problem Summary	160
4.2	Selection Constraints	160
4.3	Horizontal Constraints	160
4.3.1	Existence Constraints	160
4.3.2	Nonexistence Constraint	161
4.3.3	Existence in a Discrete Set	161

4.3.4	Nonexistence in a Discrete Set	161
4.3.5	An Attribute Verifies a Strict Comparison Constraint	161
4.3.6	An Attribute Verifies a Soft Constraint	162
4.3.6.1	A Function of Constraints: The Overall Distance	163
4.3.6.2	Strict Constraint	164
4.3.6.3	Soft Constraint	164
4.3.7	Minimum Difference Constraint	171
4.3.8	Alphanumeric Inclusion	171
4.3.9	Alphanumeric Inclusion with “Wild Card Characters”	172
4.3.10	Constraints on Virtual Attributes	172
4.3.11	Constraints on Integer Attributes	173
4.4	Vertical Constraints	173
4.4.1	Vertical Functions	174
4.4.2	Position Constraints	175
4.4.3	Soft Vertical Constraints	176
4.5	Selection Using Relational Algebra	176
4.6	Pattern Recognition	178
4.6.1	Definition and Application Areas	178
4.6.2	Extended Database Structures	179
4.6.3	Search Constraints	180
4.6.4	Database Manipulation for Pattern Recognition	180
4.6.5	Structure Definition Language (SDL) Concepts	182
4.6.5.1	Element	182
4.6.5.2	Domain	183
4.6.5.3	Structure	184
4.6.5.4	Link	184
4.6.5.5	Rule	185
4.6.5.6	Law	187
4.6.6	Image Pattern Recognition	187
4.6.6.1	Global Recognition	188
4.6.6.2	Detailed Recognition	188
4.6.6.3	Light Gradient Technique	188
4.6.6.4	Pattern Recognition with Varying Positions	189

5 DBMS TECHNIQUES AND DATABASE ARCHITECTURES

5.1	Features and Functions of a DBMS	191
5.1.1	Disk Space Management	192
5.1.2	File Access Management	193
5.1.3	File Linking	195
5.1.3.1	One-to-One (1-TO-1) Relationships	196
5.1.3.2	One-to-Many (1-TO-N) Relationships	196
5.1.3.3	Many-to-Many (N-TO-P) Relationships	200
5.1.4	Data Dictionary Management	200
5.1.5	Application Program Interface	202
5.1.5.1	Application Development Language Interface	203
5.1.5.2	Standard DBMS Operations	208
5.1.5.3	Insertion: Also Called Record Creation or Record Addition	208
5.1.5.4	Modification: Also Called Field Update	209
5.1.5.5	Deletion	213
5.1.5.6	Link Creation and Suppression	214
5.1.6	Program/Data Independence through Mapping	215
5.1.7	Program/Structure Independence Using Multilevel Views	216
5.1.8	Access Conflict and Deadlock Protection	217
5.1.8.1	Access Locking	217
5.1.8.2	Deadlocks and Dynamic Backout	218
5.1.8.3	Undoing Multiple Transactions: Committing	219
5.1.9	Backup and Recovery	222
5.1.9.1	Cold Restart	223
5.1.9.2	Warm Restart	223
5.1.10	Data Restructuring Capabilities	224
5.1.11	Security	226
5.1.12	Database Administration	227
5.1.12.1	Job Description	227
5.1.12.2	DBA Personal Profile	228
5.1.12.3	Installing the DBA	229
5.1.12.4	Administration Effort, Budget, and Reporting	229
5.1.13	Application Portability	230

5.1.14	Data Query	232
5.1.15	Evolution Capabilities	235
5.2	File Structures	236
5.2.1	Files, Records, and Fields	236
5.2.2	Segments, Blocks, Buffers, and Areas	238
5.2.3	DBMS-to-Operating System File Interface	239
5.2.4	Sequential File	240
5.2.5	Direct (Random) Access File	242
5.2.6	Index Structures	243
5.2.6.1	Primary and Secondary Indexes	245
5.2.6.2	Sequential Index: Implementation Issues	246
5.2.6.3	Hierarchical Index: Optimizing Index Block Size	247
5.2.6.4	Structure Consequences of Index Updating	250
5.2.6.5	Indexing Performance Problems	254
5.2.7	Hash-Coding	256
5.2.7.1	Principle of Hash-Coding	256
5.2.7.2	Synonyms	257
5.2.7.3	Overcoming Collision Problems	258
5.2.8	Bit-Inverted Files	265
5.2.9	List Structures: One-Way, Two-Way Rings	268
5.2.10	Dichotomy (Binary Search)	272
5.2.10.1	Binary Tree (B-Tree)	273
5.3	One-to-Many (1-TO-N) Relationships and Links	275
5.3.1	Pointer + List	276
5.3.2	Secondary Indexing	278
5.3.3	Pointer Arrays	280
5.3.4	Hash-Coded Links	281
5.4	Extended Databases: Unstructured, Open Content	282
5.4.1	Purpose of an Extended Database	282
5.4.2	Definition of an Extended Database	283
5.4.3	Topics Discussed in This Section	284

5.4.4	Definition and Physical Storage of Data, Links, and Rules	284
5.4.4.1	File Space Allocation	284
5.4.4.2	Record Management: File System Hierarchy	285
5.4.4.3	Definition and Representation of Links: Vector Function	287
5.4.4.4	Definition and Representation of Rules and Laws	288
5.4.4.5	Storage of Structures	289
5.4.5	Access by Element Content	289
5.4.6	Access by Structure (File Level, Database Level)	290
5.4.7	Access by Structure and Content	291
5.4.8	Access Structure for Links and Lists	292
5.4.9	Access Structure for Rules	292
5.4.10	Structure Definition Language (SDL)	293
5.4.10.1	Preliminary Definitions	293
5.4.10.2	Element	294
5.4.10.3	First Conclusions	296
5.4.10.4	Structure	296
5.4.10.5	Link	298
5.4.10.6	Key	298
5.4.10.7	Rule	299
5.4.11	Principles of Unstructured Data Recognition and Manipulation	299
6	OPTIMAL DATABASE IMPLEMENTATION TECHNIQUES	
6.1	Computer Load Evaluation	300
6.1.1	Purpose	300
6.1.2	Benchmarking DBMS Performance	300
6.1.2.1	Average Path Length	300
6.1.2.2	Load of Each Type of DBMS Call	301
6.1.2.3	Estimating the Average DBMS Call Execution Time	302
6.1.3	Evaluating Computer Load from Specifications	304
6.1.3.1	Number of DBMS Calls per File at Peak Usage Hour	304

6.1.3.2	Total DBMS Load at Peak Usage Hour	305
6.1.3.3	Simulating the DBMS Response Times	306
6.1.3.4	Simulation Using a Running Model of the Application	308
6.2	Optimizing Access and Linking Structures	314
6.2.1	The Dynamic Schema (the Fourth Schema) of a Database	314
6.2.2	Qualitative Architecture Decisions	314
6.2.2.1	Deriving Physical Files from Logical Files and Accesses	315
6.2.2.2	Areas and Clusters	317
6.2.2.3	Situations Where a Sequential Index Is Required	318
6.2.2.4	Situations Where Direct Access Is Possible	319
6.2.2.5	Situations Where a Bit Index Is Required	319
6.2.2.6	Direct and Indirect Hashing	319
6.2.2.7	File Linking Decisions	320
6.2.3	Quantitative Architecture Decisions	320
6.2.3.1	Principles	320
6.2.3.2	Divide the Database into Subdatabases	321
6.2.3.3	Decide Which Access and Link Paths Are Explicit/Implicit	322
6.2.3.4	Calculate the Impact of Various Alternatives	323
6.3	The Functional Interface	326
6.3.1	Principle	326
6.3.2	Architecture	326
6.3.2.1	Modules of the FI	326
6.3.2.2	Application Program Interface (API)	328
6.3.2.3	The Sequencer	331
6.3.2.4	Attribute-Level Co-Routines	336
6.3.2.5	Relation-Level Co-Routines	337
6.3.2.6	Database-Level Co-Routines	337
6.3.2.7	Interpretive Relational DBMS Interface	338
6.3.2.8	Interfacing with Nonrelational, Noninterpretive DBMSs	339
6.3.2.9	The Views Catalog	340
6.3.2.10	Multitasking, Uninterruptibility	341

6.3.2.11	Functional Journaling	343
6.3.2.12	Data Representation, Packing, and Encryption	343
6.3.2.13	Temporary Work Files	344
6.3.2.14	Virtual Attributes	344
6.3.2.15	The Functional Interface as a Package	345
6.3.3	Performance Issues	345
6.3.4	Conclusion: Pros and Cons of the FI Technique	346
6.3.5	Limited Functional Interfaces	348
6.4	The Decision Machine	349
6.4.1	Purpose	349
6.4.2	Principle	349
6.4.2.1	Parameters and Consequences of a Program Starting Decision	349
6.4.2.2	Schematic Operation of a Decision Machine	350
6.4.3	Architecture	351
6.4.3.1	Processing History File	351
6.4.3.2	Authorization File	352
6.4.3.3	Startup Queue	353
6.4.3.4	Startup Rules File	353
6.4.3.5	Rules Language	354
6.4.3.6	Modules of the Decision Machine	357
7	RELATIONAL DBMS ORACLE (Product Review of Version 4: April 1985)	
7.1	Product Description	358
7.1.1	Relational Model	358
7.1.2	The SEQUEL Language	359
7.1.3	Completeness	360
7.1.4	Physical Model	360
7.1.5	Data Dictionary	361
7.1.6	Programmer Workbench	362
7.1.7	Application Language Interface	362
7.1.8	Direct Keyboard Interface	363
7.1.9	Portability	363

7.1.10	Linking Two ORACLE Sites: IMPORT/EXPORT and ORALINK	364
7.1.11	Documentation and Learning	364
7.2	The Case Study	365
7.2.1	Functional Specifications of the Test Application	365
7.2.2	Processing Times	366
7.2.3	Problem Areas	367
7.2.4	Limitations	368
7.2.5	Program Development Speed	369
7.3	Software License Policy and Prices	369
7.4	Conclusion	369
	Index	371

