

M. Tamer Özsu • Patrick Valduriez

Principles of Distributed Database Systems

Third Edition



Springer

Preface

It has been almost twenty years since the first edition of this book appeared, and ten years since we released the second edition. As one can imagine, in a fast changing area such as this, there have been significant changes in the intervening period. Distributed data management went from a potentially significant technology to one that is common place. The advent of the Internet and the World Wide Web have certainly changed the way we typically look at distribution. The emergence in recent years of different forms of distributed computing, exemplified by data streams and cloud computing, has regenerated interest in distributed data management. Thus, it was time for a major revision of the material.

We started to work on this edition five years ago, and it has taken quite a while to complete the work. The end result, however, is a book that has been heavily revised – while we maintained and updated the core chapters, we have also added new ones. The major changes are the following:

1. Database integration and querying is now treated in much more detail, reflecting the attention these topics have received in the community in the past decade. Chapter 4 focuses on the integration process, while Chapter 9 discusses querying over multidatabase systems.
2. The previous editions had only brief discussion of data replication protocols. This topic is now covered in a separate chapter (Chapter 13) where we provide an in-depth discussion of the protocols and how they can be integrated with transaction management.
3. Peer-to-peer data management is discussed in depth in Chapter 16. These systems have become an important and interesting architectural alternative to classical distributed database systems. Although the early distributed database systems architectures followed the peer-to-peer paradigm, the modern incarnation of these systems have fundamentally different characteristics, so they deserve in-depth discussion in a chapter of their own.
4. Web data management is discussed in Chapter 17. This is a difficult topic to cover since there is no unifying framework. We discuss various aspects

of the topic ranging from web models to search engines to distributed XML processing.

5. Earlier editions contained a chapter where we discussed “recent issues” at the time. In this edition, we again have a similar chapter (Chapter 18) where we cover stream data management and cloud computing. These topics are still in a flux and are subjects of considerable ongoing research. We highlight the issues and the potential research directions.

The resulting manuscript strikes a balance between our two objectives, namely to address new and emerging issues, and maintain the main characteristics of the book in addressing the principles of distributed data management.

The organization of the book can be divided into two major parts. The first part covers the fundamental principles of distributed data management and consist of Chapters 1 to 14. Chapter 2 in this part covers the background and can be skipped if the students already have sufficient knowledge of the relational database concepts and the computer network technology. The only part of this chapter that is essential is Example 2.3, which introduces the running example that we use throughout much of the book. The second part covers more advanced topics and includes Chapters 15 – 18. What one covers in a course depends very much on the duration and the course objectives. If the course aims to discuss the fundamental techniques, then it might cover Chapters 1, 3, 5, 6–8, 10–12. An extended coverage would include, in addition to the above, Chapters 4, 9, and 13. Courses that have time to cover more material can selectively pick one or more of Chapters 15 – 18 from the second part.

Many colleagues have assisted with this edition of the book. S. Keshav (University of Waterloo) has read and provided many suggestions to update the sections on computer networks. Renée Miller (University of Toronto) and Erhard Rahm (University of Leipzig) read an early draft of Chapter 4 and provided many comments, Alon Halevy (Google) answered a number of questions about this chapter and provided a draft copy of his upcoming book on this topic as well as reading and providing feedback on Chapter 9, Avigdor Gal (Technion) also reviewed and critiqued this chapter very thoroughly. Matthias Jarke and Xiang Li (University of Aachen), Gottfried Vossen (University of Muenster), Erhard Rahm and Andreas Thor (University of Leipzig) contributed exercises to this chapter. Hubert Naacke (University of Paris 6) contributed to the section on heterogeneous cost modeling and Fabio Porto (LNCC, Petropolis) to the section on adaptive query processing of Chapter 9. Data replication (Chapter 13) could not have been written without the assistance of Gustavo Alonso (ETH Zürich) and Bettina Kemme (McGill University). Tamer spent four months in Spring 2006 visiting Gustavo where work on this chapter began and involved many long discussions. Bettina read multiple iterations of this chapter over the next one year criticizing everything and pointing out better ways of explaining the material. Esther Pacitti (University of Montpellier) also contributed to this chapter, both by reviewing it and by providing background material; she also contributed to the section on replication in database clusters in Chapter 14. Ricardo Jimenez-Peris also contributed to that chapter in the section on fault-tolerance in database clusters. Khuzaima Daudjee (University of Waterloo) read and provided

comments on this chapter as well. Chapter 15 on Distributed Object Database Management was reviewed by Serge Abiteboul (INRIA), who provided important critique of the material and suggestions for its improvement. Peer-to-peer data management (Chapter 16) owes a lot to discussions with Beng Chin Ooi (National University of Singapore) during the four months Tamer was visiting NUS in the fall of 2006. The section of Chapter 16 on query processing in P2P systems uses material from the PhD work of Reza Akbarinia (INRIA) and Wenceslao Palma (PUC-Valparaiso, Chile) while the section on replication uses material from the PhD work of Vidal Martins (PUCPR, Curitiba). The distributed XML processing section of Chapter 17 uses material from the PhD work of Ning Zhang (Facebook) and Patrick Kling at the University of Waterloo, and Ying Zhang at CWI. All three of them also read the material and provided significant feedback. Victor Muntés i Mulero (Universitat Politècnica de Catalunya) contributed to the exercises in that chapter. Özgür Ulusoy (Bilkent University) provided comments and corrections on Chapters 16 and 17. Data stream management section of Chapter 18 draws from the PhD work of Lukasz Golab (AT&T Labs-Research), and Yingying Tao at the University of Waterloo. Walid Aref (Purdue University) and Avigdor Gal (Technion) used the draft of the book in their courses, which was very helpful in debugging certain parts. We thank them, as well as many colleagues who had helped out with the first two editions, for all their assistance. We have not always followed their advice, and, needless to say, the resulting problems and errors are ours. Students in two courses at the University of Waterloo (Web Data Management in Winter 2005, and Internet-Scale Data Distribution in Fall 2005) wrote surveys as part of their coursework that were very helpful in structuring some chapters. Tamer taught courses at ETH Zürich (PDDBS – Parallel and Distributed Databases in Spring 2006) and at NUS (CS5225 – Parallel and Distributed Database Systems in Fall 2010) using parts of this edition. We thank students in all these courses for their contributions and their patience as they had to deal with chapters that were works-in-progress – the material got cleaned considerably as a result of these teaching experiences.

You will note that the publisher of the third edition of the book is different than the first two editions. Pearson, our previous publisher, decided not to be involved with the third edition. Springer subsequently showed considerable interest in the book. We would like to thank Susan Lagerstrom-Fife and Jennifer Evans of Springer for their lightning-fast decision to publish the book, and Jennifer Mauer for a ton of hand-holding during the conversion process. We would also like to thank Tracy Dunkelberger of Pearson who shepherded the reversal of the copyright to us without delay.

As in earlier editions, we will have presentation slides that can be used to teach from the book as well as solutions to most of the exercises. These will be available from Springer to instructors who adopt the book and there will be a link to them from the book's site at springer.com.

Finally, we would be very interested to hear your comments and suggestions regarding the material. We welcome any feedback, but we would particularly like to receive feedback on the following aspects:

1. any errors that may have remained despite our best efforts (although we hope there are not many);
2. any topics that should no longer be included and any topics that should be added or expanded; and
3. any exercises that you may have designed that you would like to be included in the book.

M. Tamer Özsu (Tamer.Ozsu@uwaterloo.ca)
Patrick Valduriez (Patrick.Valduriez@inria.fr)

November 2010

Contents

1	Introduction	1
1.1	Distributed Data Processing	2
1.2	What is a Distributed Database System?	3
1.3	Data Delivery Alternatives	5
1.4	Promises of DDBSs	7
1.4.1	Transparent Management of Distributed and Replicated Data	7
1.4.2	Reliability Through Distributed Transactions	12
1.4.3	Improved Performance	14
1.4.4	Easier System Expansion	15
1.5	Complications Introduced by Distribution	16
1.6	Design Issues	16
1.6.1	Distributed Database Design	17
1.6.2	Distributed Directory Management	17
1.6.3	Distributed Query Processing	17
1.6.4	Distributed Concurrency Control	18
1.6.5	Distributed Deadlock Management	18
1.6.6	Reliability of Distributed DBMS	18
1.6.7	Replication	19
1.6.8	Relationship among Problems	19
1.6.9	Additional Issues	20
1.7	Distributed DBMS Architecture	21
1.7.1	ANSI/SPARC Architecture	21
1.7.2	A Generic Centralized DBMS Architecture	23
1.7.3	Architectural Models for Distributed DBMSs	25
1.7.4	Autonomy	25
1.7.5	Distribution	27
1.7.6	Heterogeneity	27
1.7.7	Architectural Alternatives	28
1.7.8	Client/Server Systems	28
1.7.9	Peer-to-Peer Systems	32
1.7.10	Multidatabase System Architecture	35

- 1.8 Bibliographic Notes 38
- 2 Background 41**
 - 2.1 Overview of Relational DBMS 41
 - 2.1.1 Relational Database Concepts 41
 - 2.1.2 Normalization 43
 - 2.1.3 Relational Data Languages 45
 - 2.2 Review of Computer Networks 58
 - 2.2.1 Types of Networks 60
 - 2.2.2 Communication Schemes 63
 - 2.2.3 Data Communication Concepts 65
 - 2.2.4 Communication Protocols 67
 - 2.3 Bibliographic Notes 70
- 3 Distributed Database Design 71**
 - 3.1 Top-Down Design Process 73
 - 3.2 Distribution Design Issues 75
 - 3.2.1 Reasons for Fragmentation 75
 - 3.2.2 Fragmentation Alternatives 76
 - 3.2.3 Degree of Fragmentation 77
 - 3.2.4 Correctness Rules of Fragmentation 79
 - 3.2.5 Allocation Alternatives 79
 - 3.2.6 Information Requirements 80
 - 3.3 Fragmentation 81
 - 3.3.1 Horizontal Fragmentation 81
 - 3.3.2 Vertical Fragmentation 98
 - 3.3.3 Hybrid Fragmentation 112
 - 3.4 Allocation 113
 - 3.4.1 Allocation Problem 114
 - 3.4.2 Information Requirements 116
 - 3.4.3 Allocation Model 118
 - 3.4.4 Solution Methods 121
 - 3.5 Data Directory 122
 - 3.6 Conclusion 123
 - 3.7 Bibliographic Notes 125
- 4 Database Integration 131**
 - 4.1 Bottom-Up Design Methodology 133
 - 4.2 Schema Matching 137
 - 4.2.1 Schema Heterogeneity 140
 - 4.2.2 Linguistic Matching Approaches 141
 - 4.2.3 Constraint-based Matching Approaches 143
 - 4.2.4 Learning-based Matching 145
 - 4.2.5 Combined Matching Approaches 146
 - 4.3 Schema Integration 147

- 4.4 Schema Mapping 149
 - 4.4.1 Mapping Creation 150
 - 4.4.2 Mapping Maintenance 155
- 4.5 Data Cleaning 157
- 4.6 Conclusion 159
- 4.7 Bibliographic Notes 160
- 5 Data and Access Control 171**
 - 5.1 View Management 172
 - 5.1.1 Views in Centralized DBMSs 172
 - 5.1.2 Views in Distributed DBMSs 175
 - 5.1.3 Maintenance of Materialized Views 177
 - 5.2 Data Security 180
 - 5.2.1 Discretionary Access Control 181
 - 5.2.2 Multilevel Access Control 183
 - 5.2.3 Distributed Access Control 185
 - 5.3 Semantic Integrity Control 187
 - 5.3.1 Centralized Semantic Integrity Control 189
 - 5.3.2 Distributed Semantic Integrity Control 194
 - 5.4 Conclusion 200
 - 5.5 Bibliographic Notes 201
- 6 Overview of Query Processing 205**
 - 6.1 Query Processing Problem 206
 - 6.2 Objectives of Query Processing 209
 - 6.3 Complexity of Relational Algebra Operations 210
 - 6.4 Characterization of Query Processors 211
 - 6.4.1 Languages 212
 - 6.4.2 Types of Optimization 212
 - 6.4.3 Optimization Timing 213
 - 6.4.4 Statistics 213
 - 6.4.5 Decision Sites 214
 - 6.4.6 Exploitation of the Network Topology 214
 - 6.4.7 Exploitation of Replicated Fragments 215
 - 6.4.8 Use of Semijoins 215
 - 6.5 Layers of Query Processing 215
 - 6.5.1 Query Decomposition 216
 - 6.5.2 Data Localization 217
 - 6.5.3 Global Query Optimization 218
 - 6.5.4 Distributed Query Execution 219
 - 6.6 Conclusion 219
 - 6.7 Bibliographic Notes 220

7	Query Decomposition and Data Localization	221
7.1	Query Decomposition	222
7.1.1	Normalization	222
7.1.2	Analysis	223
7.1.3	Elimination of Redundancy	226
7.1.4	Rewriting	227
7.2	Localization of Distributed Data	231
7.2.1	Reduction for Primary Horizontal Fragmentation	232
7.2.2	Reduction for Vertical Fragmentation	235
7.2.3	Reduction for Derived Fragmentation	237
7.2.4	Reduction for Hybrid Fragmentation	238
7.3	Conclusion	241
7.4	Bibliographic NOTES	241
8	Optimization of Distributed Queries	245
8.1	Query Optimization	246
8.1.1	Search Space	246
8.1.2	Search Strategy	248
8.1.3	Distributed Cost Model	249
8.2	Centralized Query Optimization	257
8.2.1	Dynamic Query Optimization	257
8.2.2	Static Query Optimization	261
8.2.3	Hybrid Query Optimization	265
8.3	Join Ordering in Distributed Queries	267
8.3.1	Join Ordering	267
8.3.2	Semijoin Based Algorithms	269
8.3.3	Join versus Semijoin	272
8.4	Distributed Query Optimization	273
8.4.1	Dynamic Approach	274
8.4.2	Static Approach	277
8.4.3	Semijoin-based Approach	281
8.4.4	Hybrid Approach	286
8.5	Conclusion	290
8.6	Bibliographic Notes	292
9	Multidatabase Query Processing	297
9.1	Issues in Multidatabase Query Processing	298
9.2	Multidatabase Query Processing Architecture	299
9.3	Query Rewriting Using Views	301
9.3.1	Datalog Terminology	301
9.3.2	Rewriting in GAV	302
9.3.3	Rewriting in LAV	304
9.4	Query Optimization and Execution	307
9.4.1	Heterogeneous Cost Modeling	307
9.4.2	Heterogeneous Query Optimization	314

- 9.4.3 Adaptive Query Processing 320
- 9.5 Query Translation and Execution 327
- 9.6 Conclusion 330
- 9.7 Bibliographic Notes 331
- 10 Introduction to Transaction Management 335**
 - 10.1 Definition of a Transaction 337
 - 10.1.1 Termination Conditions of Transactions 339
 - 10.1.2 Characterization of Transactions 340
 - 10.1.3 Formalization of the Transaction Concept 341
 - 10.2 Properties of Transactions 344
 - 10.2.1 Atomicity 344
 - 10.2.2 Consistency 345
 - 10.2.3 Isolation 346
 - 10.2.4 Durability 349
 - 10.3 Types of Transactions 349
 - 10.3.1 Flat Transactions 351
 - 10.3.2 Nested Transactions 352
 - 10.3.3 Workflows 353
 - 10.4 Architecture Revisited 356
 - 10.5 Conclusion 357
 - 10.6 Bibliographic Notes 358
- 11 Distributed Concurrency Control 361**
 - 11.1 Serializability Theory 362
 - 11.2 Taxonomy of Concurrency Control Mechanisms 367
 - 11.3 Locking-Based Concurrency Control Algorithms 369
 - 11.3.1 Centralized 2PL 373
 - 11.3.2 Distributed 2PL 374
 - 11.4 Timestamp-Based Concurrency Control Algorithms 377
 - 11.4.1 Basic TO Algorithm 378
 - 11.4.2 Conservative TO Algorithm 381
 - 11.4.3 Multiversion TO Algorithm 383
 - 11.5 Optimistic Concurrency Control Algorithms 384
 - 11.6 Deadlock Management 387
 - 11.6.1 Deadlock Prevention 389
 - 11.6.2 Deadlock Avoidance 390
 - 11.6.3 Deadlock Detection and Resolution 391
 - 11.7 “Relaxed” Concurrency Control 394
 - 11.7.1 Non-Serializable Histories 395
 - 11.7.2 Nested Distributed Transactions 396
 - 11.8 Conclusion 398
 - 11.9 Bibliographic Notes 401

12	Distributed DBMS Reliability	405
12.1	Reliability Concepts and Measures	406
12.1.1	System, State, and Failure	406
12.1.2	Reliability and Availability	408
12.1.3	Mean Time between Failures/Mean Time to Repair	409
12.2	Failures in Distributed DBMS	410
12.2.1	Transaction Failures	411
12.2.2	Site (System) Failures	411
12.2.3	Media Failures	412
12.2.4	Communication Failures	412
12.3	Local Reliability Protocols	413
12.3.1	Architectural Considerations	413
12.3.2	Recovery Information	416
12.3.3	Execution of LRM Commands	420
12.3.4	Checkpointing	425
12.3.5	Handling Media Failures	426
12.4	Distributed Reliability Protocols	427
12.4.1	Components of Distributed Reliability Protocols	428
12.4.2	Two-Phase Commit Protocol	428
12.4.3	Variations of 2PC	434
12.5	Dealing with Site Failures	436
12.5.1	Termination and Recovery Protocols for 2PC	437
12.5.2	Three-Phase Commit Protocol	443
12.6	Network Partitioning	448
12.6.1	Centralized Protocols	450
12.6.2	Voting-based Protocols	450
12.7	Architectural Considerations	453
12.8	Conclusion	454
12.9	Bibliographic Notes	455
13	Data Replication	459
13.1	Consistency of Replicated Databases	461
13.1.1	Mutual Consistency	461
13.1.2	Mutual Consistency versus Transaction Consistency	463
13.2	Update Management Strategies	465
13.2.1	Eager Update Propagation	465
13.2.2	Lazy Update Propagation	466
13.2.3	Centralized Techniques	466
13.2.4	Distributed Techniques	467
13.3	Replication Protocols	468
13.3.1	Eager Centralized Protocols	468
13.3.2	Eager Distributed Protocols	474
13.3.3	Lazy Centralized Protocols	475
13.3.4	Lazy Distributed Protocols	480
13.4	Group Communication	482

- 13.5 Replication and Failures 485
 - 13.5.1 Failures and Lazy Replication 485
 - 13.5.2 Failures and Eager Replication 486
- 13.6 Replication Mediator Service 489
- 13.7 Conclusion 491
- 13.8 Bibliographic Notes 493

- 14 Parallel Database Systems 497**
 - 14.1 Parallel Database System Architectures 498
 - 14.1.1 Objectives 498
 - 14.1.2 Functional Architecture 501
 - 14.1.3 Parallel DBMS Architectures 502
 - 14.2 Parallel Data Placement 508
 - 14.3 Parallel Query Processing 512
 - 14.3.1 Query Parallelism 513
 - 14.3.2 Parallel Algorithms for Data Processing 515
 - 14.3.3 Parallel Query Optimization 521
 - 14.4 Load Balancing 525
 - 14.4.1 Parallel Execution Problems 525
 - 14.4.2 Intra-Operator Load Balancing 527
 - 14.4.3 Inter-Operator Load Balancing 529
 - 14.4.4 Intra-Query Load Balancing 530
 - 14.5 Database Clusters 534
 - 14.5.1 Database Cluster Architecture 535
 - 14.5.2 Replication 537
 - 14.5.3 Load Balancing 540
 - 14.5.4 Query Processing 542
 - 14.5.5 Fault-tolerance 545
 - 14.6 Conclusion 546
 - 14.7 Bibliographic Notes 547

- 15 Distributed Object Database Management 551**
 - 15.1 Fundamental Object Concepts and Object Models 553
 - 15.1.1 Object 553
 - 15.1.2 Types and Classes 556
 - 15.1.3 Composition (Aggregation) 557
 - 15.1.4 Subclassing and Inheritance 558
 - 15.2 Object Distribution Design 560
 - 15.2.1 Horizontal Class Partitioning 561
 - 15.2.2 Vertical Class Partitioning 563
 - 15.2.3 Path Partitioning 563
 - 15.2.4 Class Partitioning Algorithms 564
 - 15.2.5 Allocation 565
 - 15.2.6 Replication 565
 - 15.3 Architectural Issues 566

15.3.1	Alternative Client/Server Architectures	567
15.3.2	Cache Consistency	572
15.4	Object Management	574
15.4.1	Object Identifier Management	574
15.4.2	Pointer Swizzling	576
15.4.3	Object Migration	577
15.5	Distributed Object Storage	578
15.6	Object Query Processing	582
15.6.1	Object Query Processor Architectures	583
15.6.2	Query Processing Issues	584
15.6.3	Query Execution	589
15.7	Transaction Management	593
15.7.1	Correctness Criteria	594
15.7.2	Transaction Models and Object Structures	596
15.7.3	Transactions Management in Object DBMSs	596
15.7.4	Transactions as Objects	605
15.8	Conclusion	606
15.9	Bibliographic Notes	607
16	Peer-to-Peer Data Management	611
16.1	Infrastructure	614
16.1.1	Unstructured P2P Networks	615
16.1.2	Structured P2P Networks	618
16.1.3	Super-peer P2P Networks	622
16.1.4	Comparison of P2P Networks	624
16.2	Schema Mapping in P2P Systems	624
16.2.1	Pairwise Schema Mapping	625
16.2.2	Mapping based on Machine Learning Techniques	626
16.2.3	Common Agreement Mapping	626
16.2.4	Schema Mapping using IR Techniques	627
16.3	Querying Over P2P Systems	628
16.3.1	Top-k Queries	628
16.3.2	Join Queries	640
16.3.3	Range Queries	642
16.4	Replica Consistency	645
16.4.1	Basic Support in DHTs	646
16.4.2	Data Currency in DHTs	648
16.4.3	Replica Reconciliation	649
16.5	Conclusion	653
16.6	Bibliographic Notes	653
17	Web Data Management	657
17.1	Web Graph Management	658
17.1.1	Compressing Web Graphs	660
17.1.2	Storing Web Graphs as S-Nodes	661

- 17.2 Web Search 663
 - 17.2.1 Web Crawling 664
 - 17.2.2 Indexing 667
 - 17.2.3 Ranking and Link Analysis 668
 - 17.2.4 Evaluation of Keyword Search 669
- 17.3 Web Querying 670
 - 17.3.1 Semistructured Data Approach 671
 - 17.3.2 Web Query Language Approach 676
 - 17.3.3 Question Answering 681
 - 17.3.4 Searching and Querying the Hidden Web 685
- 17.4 Distributed XML Processing 689
 - 17.4.1 Overview of XML 691
 - 17.4.2 XML Query Processing Techniques 699
 - 17.4.3 Fragmenting XML Data 703
 - 17.4.4 Optimizing Distributed XML Processing 710
- 17.5 Conclusion 718
- 17.6 Bibliographic Notes 719

- 18 Current Issues: Streaming Data and Cloud Computing 723**
 - 18.1 Data Stream Management 723
 - 18.1.1 Stream Data Models 725
 - 18.1.2 Stream Query Languages 727
 - 18.1.3 Streaming Operators and their Implementation 732
 - 18.1.4 Query Processing 734
 - 18.1.5 DSMS Query Optimization 738
 - 18.1.6 Load Shedding and Approximation 739
 - 18.1.7 Multi-Query Optimization 740
 - 18.1.8 Stream Mining 741
 - 18.2 Cloud Data Management 744
 - 18.2.1 Taxonomy of Clouds 745
 - 18.2.2 Grid Computing 748
 - 18.2.3 Cloud architectures 751
 - 18.2.4 Data management in the cloud 753
 - 18.3 Conclusion 760
 - 18.4 Bibliographic Notes 762

- References 765**

- Index 833**