

# CONTENTS

## PREFACE

xvii

## 1 INTRODUCTION

1

- 1.1 DEFINITION OF A DISTRIBUTED SYSTEM 2
- 1.2 GOALS 4
  - 1.2.1 Connecting Users and Resources 4
  - 1.2.2 Transparency 5
  - 1.2.3 Openness 8
  - 1.2.4 Scalability 10
- 1.3 HARDWARE CONCEPTS 16
  - 1.3.1 Multiprocessors 17
  - 1.3.2 Homogeneous Multicomputer Systems 19
  - 1.3.3 Heterogeneous Multicomputer Systems 21
- 1.4 SOFTWARE CONCEPTS 22
  - 1.4.1 Distributed Operating Systems 22
  - 1.4.2 Network Operating Systems 33
  - 1.4.3 Middleware 36
- 1.5 THE CLIENT-SERVER MODEL 42
  - 1.5.1 Clients and Servers 42
  - 1.5.2 Application Layering 46
  - 1.5.3 Client-Server Architectures 50
- 1.6 SUMMARY 53

## 2 COMMUNICATION

57

- 2.1 LAYERED PROTOCOLS 58
  - 2.1.1 Lower-Level Protocols 61
  - 2.1.2 Transport Protocols 63
  - 2.1.3 Higher-Level Protocols 66

vii

2.2	REMOTE PROCEDURE CALL	68
2.2.1	Basic RPC Operation	69
2.2.2	Parameter Passing	73
2.2.3	Extended RPC Models	77
2.2.4	Example: DCE RPC	80
2.3	REMOTE OBJECT INVOCATION	85
2.3.1	Distributed Objects	86
2.3.2	Binding a Client to an Object	88
2.3.3	Static versus Dynamic Remote Method Invocations	90
2.3.4	Parameter Passing	91
2.3.5	Example 1: DCE Remote Objects	93
2.3.6	Example 2: Java RMI	95
2.4	MESSAGE-ORIENTED COMMUNICATION	99
2.4.1	Persistence and Synchronicity in Communication	99
2.4.2	Message-Oriented Transient Communication	104
2.4.3	Message-Oriented Persistent Communication	108
2.4.4	Example: IBM MQSeries	115
2.5	STREAM-ORIENTED COMMUNICATION	119
2.5.1	Support for Continuous Media	120
2.5.2	Streams and Quality of Service	123
2.5.3	Stream Synchronization	127
2.6	SUMMARY	130
<b>3</b>	<b>PROCESSES</b>	<b>135</b>
3.1	THREADS	136
3.1.1	Introduction to Threads	136
3.1.2	Threads in Distributed Systems	141
3.2	CLIENTS	145
3.2.1	User Interfaces	145
3.2.2	Client-Side Software for Distribution Transparency	147
3.3	SERVERS	149
3.3.1	General Design Issues	149
3.3.2	Object Servers	152
3.4	CODE MIGRATION	158
3.4.1	Approaches to Code Migration	158
3.4.2	Migration and Local Resources	163
3.4.3	Migration in Heterogeneous Systems	165
3.4.4	Example: D'Agents	168

3.5	SOFTWARE AGENTS	173
3.5.1	Software Agents in Distributed Systems	173
3.5.2	Agent Technology	175
3.6	SUMMARY	178

**4 NAMING****183**

4.1	NAMING ENTITIES	184
4.1.1	Names, Identifiers, and Addresses	184
4.1.2	Name Resolution	189
4.1.3	The Implementation of a Name Space	194
4.1.4	Example: The Domain Name System	201
4.1.5	Example: X.500	206
4.2	LOCATING MOBILE ENTITIES	210
4.2.1	Naming versus Locating Entities	210
4.2.2	Simple Solutions	212
4.2.3	Home-Based Approaches	216
4.2.4	Hierarchical Approaches	217
4.3	REMOVING UNREFERENCED ENTITIES	225
4.3.1	The Problem of Unreferenced Objects	225
4.3.2	Reference Counting	227
4.3.3	Reference Listing	231
4.3.4	Identifying Unreachable Entities	232
4.4	SUMMARY	238

**5 SYNCHRONIZATION****241**

5.1	CLOCK SYNCHRONIZATION	242
5.1.1	Physical Clocks	243
5.1.2	Clock Synchronization Algorithms	246
5.1.3	Use of Synchronized Clocks	251
5.2	LOGICAL CLOCKS	252
5.2.1	Lamport timestamps	252
5.2.2	Vector timestamps	256
5.3	GLOBAL STATE	258
5.4	ELECTION ALGORITHMS	262
5.4.1	The Bully Algorithm	262
5.4.2	A Ring Algorithm	263
5.5	MUTUAL EXCLUSION	265
5.5.1	A Centralized Algorithm	265
5.5.2	A Distributed Algorithm	266
5.5.3	A Token Ring Algorithm	269
5.5.4	A Comparison of the Three Algorithms	270

- 5.6 DISTRIBUTED TRANSACTIONS 271
    - 5.6.1 The Transaction Model 272
    - 5.6.2 Classification of Transactions 275
    - 5.6.3 Implementation 278
    - 5.6.4 Concurrency Control 280
  - 5.7 SUMMARY 288
- 6 CONSISTENCY AND REPLICATION 291**
- 6.1 INTRODUCTION 292
    - 6.1.1 Reasons for Replication 292
    - 6.1.2 Object Replication 293
    - 6.1.3 Replication as Scaling Technique 296
  - 6.2 DATA-CENTRIC CONSISTENCY MODELS 297
    - 6.2.1 Strict Consistency 298
    - 6.2.2 Linearizability and Sequential Consistency 300
    - 6.2.3 Causal Consistency 305
    - 6.2.4 FIFO Consistency 306
    - 6.2.5 Weak Consistency 308
    - 6.2.6 Release Consistency 310
    - 6.2.7 Entry Consistency 313
    - 6.2.8 Summary of Consistency Models 315
  - 6.3 CLIENT-CENTRIC CONSISTENCY MODELS 316
    - 6.3.1 Eventual Consistency 317
    - 6.3.2 Monotonic Reads 319
    - 6.3.3 Monotonic Writes 320
    - 6.3.4 Read Your Writes 322
    - 6.3.5 Writes Follow Reads 323
    - 6.3.6 Implementation 324
  - 6.4 DISTRIBUTION PROTOCOLS 326
    - 6.4.1 Replica Placement 326
    - 6.4.2 Update Propagation 330
    - 6.4.3 Epidemic Protocols 334
  - 6.5 CONSISTENCY PROTOCOLS 337
    - 6.5.1 Primary-Based Protocols 337
    - 6.5.2 Replicated-Write Protocols 341
    - 6.5.3 Cache-Coherence Protocols 345
  - 6.6 EXAMPLES 346
    - 6.6.1 Orca 347
    - 6.6.2 Causally-Consistent Lazy Replication 352
  - 6.7 SUMMARY 357

**7 FAULT TOLERANCE 361**

- 7.1 INTRODUCTION TO FAULT TOLERANCE 362
  - 7.1.1 Basic Concepts 362
  - 7.1.2 Failure Models 364
  - 7.1.3 Failure Masking by Redundancy 366
- 7.2 PROCESS RESILIENCE 368
  - 7.2.1 Design Issues 368
  - 7.2.2 Failure Masking and Replication 370
  - 7.2.3 Agreement in Faulty Systems 371
- 7.3 RELIABLE CLIENT-SERVER COMMUNICATION 375
  - 7.3.1 Point-to-Point Communication 375
  - 7.3.2 RPC Semantics in the Presence of Failures 375
- 7.4 RELIABLE GROUP COMMUNICATION 381
  - 7.4.1 Basic Reliable-Multicasting Schemes 381
  - 7.4.2 Scalability in Reliable Multicasting 383
  - 7.4.3 Atomic Multicast 386
- 7.5 DISTRIBUTED COMMIT 393
  - 7.5.1 Two-Phase Commit 393
  - 7.5.2 Three-Phase Commit 399
- 7.6 RECOVERY 401
  - 7.6.1 Introduction 401
  - 7.6.2 Checkpointing 404
  - 7.6.3 Message Logging 407
- 7.7 SUMMARY 410

**8 SECURITY 413**

- 8.1 INTRODUCTION TO SECURITY 414
  - 8.1.1 Security Threats, Policies, and Mechanisms 414
  - 8.1.2 Design Issues 420
  - 8.1.3 Cryptography 425
- 8.2 SECURE CHANNELS 432
  - 8.2.1 Authentication 433
  - 8.2.2 Message Integrity and Confidentiality 441
  - 8.2.3 Secure Group Communication 444
- 8.3 ACCESS CONTROL 447
  - 8.3.1 General Issues in Access Control 447
  - 8.3.2 Firewalls 451
  - 8.3.3 Secure Mobile Code 453

- 8.4 SECURITY MANAGEMENT 460
  - 8.4.1 Key Management 461
  - 8.4.2 Secure Group Management 465
  - 8.4.3 Authorization Management 466
- 8.5 EXAMPLE: KERBEROS 472
- 8.6 EXAMPLE: SESAME 473
  - 8.6.1 SESAME Components 474
  - 8.6.2 Privilege Attribute Certificates (PACs) 477
- 8.7 EXAMPLE: ELECTRONIC PAYMENT SYSTEMS 478
  - 8.7.1 Electronic Payment Systems 478
  - 8.7.2 Security in Electronic Payment Systems 480
  - 8.7.3 Example Protocols 484
- 8.8 SUMMARY 488

## 9 DISTRIBUTED OBJECT-BASED SYSTEMS 493

- 9.1 CORBA 494
  - 9.1.1 Overview of CORBA 495
  - 9.1.2 Communication 501
  - 9.1.3 Processes 508
  - 9.1.4 Naming 514
  - 9.1.5 Synchronization 518
  - 9.1.6 Caching and Replication 518
  - 9.1.7 Fault Tolerance 520
  - 9.1.8 Security 522
- 9.2 DISTRIBUTED COM 525
  - 9.2.1 Overview of DCOM 526
  - 9.2.2 Communication 531
  - 9.2.3 Processes 534
  - 9.2.4 Naming 537
  - 9.2.5 Synchronization 541
  - 9.2.6 Replication 541
  - 9.2.7 Fault Tolerance 541
  - 9.2.8 Security 542
- 9.3 GLOBE 545
  - 9.3.1 Overview of Globe 545
  - 9.3.2 Communication 553
  - 9.3.3 Processes 554
  - 9.3.4 Naming 557
  - 9.3.5 Synchronization 559
  - 9.3.6 Replication 560

- 9.3.7 Fault Tolerance 563
- 9.3.8 Security 563
- 9.4 COMPARISON OF CORBA, DCOM, AND GLOBE 565
  - 9.4.1 Philosophy 566
  - 9.4.2 Communication 567
  - 9.4.3 Processes 567
  - 9.4.4 Naming 568
  - 9.4.5 Synchronization 569
  - 9.4.6 Caching and Replication 569
  - 9.4.7 Fault Tolerance 570
  - 9.4.8 Security 570
- 9.5 SUMMARY 572

## 10 DISTRIBUTED FILE SYSTEMS 575

- 10.1 SUN NETWORK FILE SYSTEM 576
  - 10.1.1 Overview of NFS 576
  - 10.1.2 Communication 581
  - 10.1.3 Processes 582
  - 10.1.4 Naming 583
  - 10.1.5 Synchronization 590
  - 10.1.6 Caching and Replication 594
  - 10.1.7 Fault Tolerance 597
  - 10.1.8 Security 600
- 10.2 THE CODA FILE SYSTEM 604
  - 10.2.1 Overview of Coda 604
  - 10.2.2 Communication 606
  - 10.2.3 Processes 608
  - 10.2.4 Naming 609
  - 10.2.5 Synchronization 610
  - 10.2.6 Caching and Replication 615
  - 10.2.7 Fault Tolerance 618
  - 10.2.8 Security 620
- 10.3 OTHER DISTRIBUTED FILE SYSTEMS 623
  - 10.3.1 Plan 9: Resources Unified to Files 623
  - 10.3.2 XFS: Serverless File System 629
  - 10.3.3 SFS: Scalable Security 635
- 10.4 COMPARISON OF DISTRIBUTED FILE SYSTEMS 638
  - 10.4.1 Philosophy 638
  - 10.4.2 Communication 639
  - 10.4.3 Processes 639

- 10.4.4 Naming 640
- 10.4.5 Synchronization 641
- 10.4.6 Caching and Replication 641
- 10.4.7 Fault Tolerance 642
- 10.4.8 Security 642
- 10.5 SUMMARY 643

## 11 DISTRIBUTED DOCUMENT-BASED SYSTEMS 647

- 11.1 THE WORLD WIDE WEB 648
  - 11.1.1 Overview of WWW 648
  - 11.1.2 Communication 657
  - 11.1.3 Processes 662
  - 11.1.4 Naming 668
  - 11.1.5 Synchronization 671
  - 11.1.6 Caching and Replication 672
  - 11.1.7 Fault Tolerance 676
  - 11.1.8 Security 676
- 11.2 LOTUS NOTES 678
  - 11.2.1 Overview of Lotus Notes 678
  - 11.2.2 Communication 680
  - 11.2.3 Processes 681
  - 11.2.4 Naming 683
  - 11.2.5 Synchronization 685
  - 11.2.6 Replication 685
  - 11.2.7 Fault Tolerance 688
  - 11.2.8 Security 688
- 11.3 COMPARISON OF WWW AND LOTUS NOTES 691
- 11.4 SUMMARY 695

## 12 DISTRIBUTED COORDINATION-BASED SYSTEMS 699

- 12.1 INTRODUCTION TO COORDINATION MODELS 700
- 12.2 TIB/RENDEZVOUS 702
  - 12.2.1 Overview of TIB/Rendezvous 702
  - 12.2.2 Communication 704
  - 12.2.3 Processes 708
  - 12.2.4 Naming 709
  - 12.2.5 Synchronization 710
  - 12.2.6 Caching and Replication 712
  - 12.2.7 Fault Tolerance 713
  - 12.2.8 Security 715

- 12.3 JINI 716
  - 12.3.1 Overview of Jini 717
  - 12.3.2 Communication 719
  - 12.3.3 Processes 721
  - 12.3.4 Naming 724
  - 12.3.5 Synchronization 727
  - 12.3.6 Caching and Replication 728
  - 12.3.7 Fault Tolerance 728
  - 12.3.8 Security 729
- 12.4 COMPARISON OF TIB/RENDEZVOUS AND JINI 730
- 12.5 SUMMARY 733

## 13 READING LIST AND BIBLIOGRAPHY 737

- 13.1 SUGGESTIONS FOR FURTHER READING 737
  - 13.1.1 Introduction and General Works 737
  - 13.1.2 Communication 739
  - 13.1.3 Processes 739
  - 13.1.4 Naming 740
  - 13.1.5 Synchronization 741
  - 13.1.6 Consistency and Replication 742
  - 13.1.7 Fault Tolerance 743
  - 13.1.8 Security 744
  - 13.1.9 Distributed Object-Based Systems 745
  - 13.1.10 Distributed File Systems 746
  - 13.1.11 Distributed Document-Based Systems 747
  - 13.1.12 Distributed Coordination-Based Systems 748
- 13.2 ALPHABETICAL BIBLIOGRAPHY 749

## INDEX

785