CONTENTS

FOREWARD v

PREFACE xi

1 CHARACTERIZATION OF DISTRIBUTED SYSTEMS 1

1.1 Introduction 2
1.2 Examples 3
1.3 Key characteristics 10
1.4 Historical background 21
1.5 Summary 25

2 DESIGN GOALS 29

2.1 Introduction 30
2.2 Basic design issues 31
2.3 User requirements 49
2.4 Summary 54

3 NETWORKING AND INTERNETWORKING 59

3.1 Introduction 60
3.2 Network technologies 66
3.3 Protocols 69
3.4 Technology case studies: Ethernet, Token Ring and ATM 75
3.5 Protocol case studies: Internet protocols and FLIP 84
3.6 Summary 95

4 INTERPROCESS COMMUNICATION 99

4.1 Introduction 100

4.2 Building blocks 101

4.3 Client-server communication 108

4.4 Group communication 112

4.5 Case study: interprocess communication in UNIX 119

4.6 Summary 123

5 REMOTE PROCEDURE CALLING 127

5.1 Introduction 128

5.2 Design issues 130

5.3 Implementation 134

5.4 Case studies: Sun and ANSA 138

5.5 Asynchronous RPC 149

5.6 Summary 152

6 DISTRIBUTED OPERATING SYSTEMS 157

6.1 Introduction 158

6.2 The kernel 160

6.3 Processes and threads 165

6.4 Naming and protection 178

6.5 Communication and invocation 182

6.6 Virtual memory 190

6.7 Summary 193

7 FILE SERVICE: A MODEL 197

7.1 Introduction 198

7.2 File service components 200

7.3 Design issues 202

7.4 Interfaces 204

7.5 Implementation techniques 210

7.6 Summary 219

8 FILE SERVICE: CASE STUDIES 221

8.1 Introduction 222
8.2 The Sun Network File System 222
8.3 The Andrew File System 233
8.4 The Coda File System 243
8.5 Summary 249

9 NAME SERVICES 253

9.1 Introduction 254
9.2 The SNS – a name service model 257
9.3 Discussion of the SNS and further design issues 264
9.4 Case studies: DNS, GNS and X.500 268
9.5 Summary 282

10 TIME AND COORDINATION 287

10.1 Introduction 288
10.2 Synchronizing physical clocks 289
10.3 Logical time and logical clocks 297
10.4 Distributed coordination 300
10.5 Summary 308

11 REPLICATION 311

11.1 Introduction 312
11.2 Basic architectural model 316
11.3 Consistency and request ordering 319
11.4 The gossip architecture 327
11.5 Process groups and ISIS 335
11.6 Summary 348

12 SHARED DATA AND TRANSACTIONS 353

12.1 Introduction 354
12.2 Conversations between a client and a server 356
12.3 Fault tolerance and recovery 358
12.4 Transactions 359
12.5 Nested transactions 370
12.6 Summary 371

13 CONCURRENCY CONTROL 377

13.1 Introduction 378

13.2 Locks 381

13.3 Optimistic concurrency control 392

13.4 Timestamp ordering 396

13.5 Comparison of methods for concurrency control 403

13.6 Summary 404

14 DISTRIBUTED TRANSACTIONS 409

14.1 Introduction 410

14.2 Simple distributed transactions and nested transactions 410

14.3 Atomic commit protocols 414

14.4 Concurrency control in distributed transactions 422

14.5 Distributed deadlocks 424

14.6 Transactions with replicated data 432

14.7 Summary 444

15 RECOVERY AND FAULT TOLERANCE 449

15.1 Introduction 450

15.2 Transaction recovery 450

15.3 Fault tolerance 461

15.4 Hierarchical and group masking of faults 466

15.5 Summary 472

16 SECURITY 477

16.1 Introduction 478

16.2 Cryptography 485

16.3 Authentication and key distribution 492

16.4 Case study: Kerberos 495

16.5 Logics of authentication 503

16.6 Digital signatures 510

16.7 Summary 513

17 DISTRIBUTED SHARED MEMORY 517

17.1 Introduction 518
17.2 Design and implementation issues 522
17.3 Sequential consistency and Ivy 529
17.4 Release consistency and Munin 537
17.5 Other consistency models 540
17.6 Summary 542

18 DISTRIBUTED OPERATING SYSTEMS: CASE STUDIES 545

18.1 Introduction 546
18.2 Mach 547
18.3 Chorus 566
18.4 UNIX emulation in Mach and Chorus 579
18.5 Amoeba 584
18.6 A comparison of Mach, Amoeba and Chorus 594
18.7 Clouds 597
18.8 Firefly RPC 603
18.9 The Amoeba multicast protocol 608

REFERENCES 617

INDEX 635