
Contents

1	Introduction	1
1.1	Green Clouds on the Horizon	1
1.2	The Cloud to the Rescue!	4
1.3	A Simple Cloud Computing Application	5
1.4	Stability and Scalability: Contending Goals in Cloud Settings . .	10
1.5	The Missing Theory of Cloud Scalability	18
1.6	Brewer’s CAP Conjecture	22
1.7	The Challenge of Trusted Computing in Cloud Settings	28
1.8	Data Replication: The Foundational Cloud Technology	35
1.9	Split Brains and Other Forms of Mechanized Insanity	39
1.10	Conclusions	42

Part I Computing in the Cloud

2	The Way of the Cloud	45
2.1	Introduction	45
2.1.1	The Technical and Social Origins of the Cloud	45
2.1.2	Is the Cloud a Distributed Computing Technology?	50
2.1.3	What Does Reliability Mean in the Cloud?	60
2.2	Components of a Reliable Distributed Computing System	63
2.3	Summary: Reliability in the Cloud	65
2.4	Related Reading	67
3	Client Perspective	69
3.1	The Life of a Cloud Computing Client	69
3.2	Web Services	70
3.2.1	How Web Browsers Talk to Web Sites	70
3.2.2	Web Services: Client/Server RPC over HTTP	76
3.3	WS_RELIABILITY and WS_SECURITY	81
3.3.1	WS_RELIABILITY	81
3.3.2	WS_SECURITY	83
3.3.3	WS_SECURITY	86
3.4	Safe Execution of Downloaded Code	87
3.5	Coping with Mobility	95
3.6	The Multicore Client	97

3.7	Conclusions	98
3.8	Further Readings	99
4	Network Perspective	101
4.1	Network Perspective	101
4.2	The Many Dimensions of Network Reliability	101
4.2.1	Internet Routers: A Rapidly Evolving Technology Arena	103
4.2.2	The Border Gateway Protocol Under Pressure	109
4.2.3	Consistency in Network Routing	115
4.2.4	Extensible Routers	116
4.2.5	Overlay Networks	118
4.2.6	RON: The Resilient Overlay Network	119
4.2.7	Distributed Hash Tables: Chord, Pastry, Beehive and Kelips	122
4.2.8	BitTorrent: A Fast Content Distribution System	136
4.2.9	Sienna: A Content-Based Publish Subscribe System	137
4.2.10	The Internet Under Attack: A Spectrum of Threats	140
4.3	Summary and Conclusions	142
4.4	Further Readings	143
5	The Structure of Cloud Data Centers	145
5.1	The Layers of a Cloud	146
5.2	Elasticity and Reconfigurability	146
5.3	Rapid Local Responsiveness and CAP	148
5.4	Heavily Skewed Workloads and Zipf's Law	151
5.5	A Closer Look at the First Tier	155
5.6	Soft State vs. Hard State	157
5.7	Services Supporting the First Tier	158
5.7.1	Memcached	158
5.7.2	BigTable	159
5.7.3	Dynamo	162
5.7.4	PNUTS and Cassandra	164
5.7.5	Chubby	165
5.7.6	Zookeeper	165
5.7.7	Sinfonia	166
5.7.8	The Smoke and Mirrors File System	167
5.7.9	Message Queuing Middleware	169
5.7.10	Cloud Management Infrastructure and Tools	172
5.8	Life in the Back	172
5.9	The Emergence of the Rent-A-Cloud Model	175
5.9.1	Can HPC Applications Run on the Cloud?	177
5.10	Issues Associated with Cloud Storage	180
5.11	Related Reading	183
6	Remote Procedure Calls and the Client/Server Model	185
6.1	Remote Procedure Call: The Foundation of Client/Server Computing	185

6.2	RPC Protocols and Concepts	188
6.3	Writing an RPC-Based Client or Server Program	191
6.4	The RPC Binding Problem	195
6.5	Marshalling and Data Types	197
6.6	Associated Services	199
6.6.1	Naming Services	200
6.6.2	Security Services	202
6.6.3	Transactions	203
6.7	The RPC Protocol	204
6.8	Using RPC in Reliable Distributed Systems	208
6.9	Layering RPC over TCP	211
6.10	Stateless and Stateful Client/Server Interactions	213
6.11	Major Uses of the Client/Server Paradigm	213
6.12	Distributed File Systems	219
6.13	Stateful File Servers	227
6.14	Distributed Database Systems	236
6.15	Applying Transactions to File Servers	243
6.16	Related Reading	245
7	CORBA: The Common Object Request Broker Architecture	249
7.1	The ANSA Project	250
7.2	Beyond ANSA to CORBA	252
7.3	The CORBA Reference Model	254
7.4	IDL and ODL	260
7.5	ORB	261
7.6	Naming Service	262
7.7	ENS—The CORBA Event Notification Service	262
7.8	Life-Cycle Service	264
7.9	Persistent Object Service	264
7.10	Transaction Service	264
7.11	Interobject Broker Protocol	264
7.12	Properties of CORBA Solutions	265
7.13	Performance of CORBA and Related Technologies	266
7.14	Related Reading	269
8	System Support for Fast Client/Server Communication	271
8.1	Lightweight RPC	271
8.2	fbufs and the <i>x</i> -Kernel Project	274
8.3	Active Messages	276
8.4	Beyond Active Messages: U-Net and the Virtual Interface Architecture (VIA)	278
8.5	Asynchronous I/O APIs	282
8.6	Related Reading	283

Part II Reliable Distributed Computing

9	How and Why Computer Systems Fail	287
9.1	Hardware Reliability and Trends	288
9.2	Software Reliability and Trends	289
9.3	Other Sources of Downtime	292
9.4	Complexity	292
9.5	Detecting Failures	294
9.6	Hostile Environments	295
9.7	Related Reading	299
10	Overcoming Failures in a Distributed System	301
10.1	Consistent Distributed Behavior	301
10.1.1	Static Membership	309
10.1.2	Dynamic Membership	313
10.2	Time in Distributed Systems	316
10.3	The Distributed Commit Problem	323
10.3.1	Two-Phase Commit	326
10.3.2	Three-Phase Commit	332
10.3.3	Quorum Update Revisited	336
10.4	Related Reading	336
11	Dynamic Membership	339
11.1	Dynamic Group Membership	339
11.1.1	GMS and Other System Processes	341
11.1.2	Protocol Used to Track GMS Membership	346
11.1.3	GMS Protocol to Handle Client Add and Join Events	348
11.1.4	GMS Notifications with Bounded Delay	349
11.1.5	Extending the GMS to Allow Partition and Merge Events	352
11.2	Replicated Data with Malicious Failures	353
11.3	The Impossibility of Asynchronous Consensus (FLP)	359
11.3.1	Three-Phase Commit and Consensus	362
11.4	Extending Our Protocol into a Full GMS	365
11.5	Related Reading	367
12	Group Communication Systems	369
12.1	Group Communication	369
12.2	A Closer Look at Delivery Ordering Options	374
12.2.1	Nondurable Failure-Atomic Group Multicast	378
12.2.2	Strongly Durable Failure-Atomic Group Multicast	380
12.2.3	Dynamic Process Groups	381
12.2.4	View-Synchronous Failure Atomicity	383
12.2.5	Summary of GMS Properties	385
12.2.6	Ordered Multicast	386
12.3	Communication from Nonmembers to a Group	399
12.4	Communication from a Group to a Nonmember	402

12.5	Summary of Multicast Properties	403
12.6	Related Reading	404
13	Point to Point and Multi-group Considerations	407
13.1	Causal Communication Outside of a Process Group	408
13.2	Extending Causal Order to Multigroup Settings	411
13.3	Extending Total Order to Multigroup Settings	413
13.4	Causal and Total Ordering Domains	415
13.5	Multicasts to Multiple Groups	416
13.6	Multigroup View Management Protocols	417
13.7	Related Reading	418
14	The Virtual Synchrony Execution Model	419
14.1	Virtual Synchrony	419
14.2	Extended Virtual Synchrony	424
14.3	Virtually Synchronous Algorithms and Tools	430
14.3.1	Replicated Data and Synchronization	430
14.3.2	State Transfer to a Joining Process	435
14.3.3	Load-Balancing	437
14.3.4	Primary-Backup Fault Tolerance	438
14.3.5	Coordinator-Cohort Fault Tolerance	440
14.3.6	Applying Virtual Synchrony in the Cloud	442
14.4	Related Reading	455
15	Consistency in Distributed Systems	457
15.1	Consistency in the Static and Dynamic Membership Models	458
15.2	Practical Options for Coping with Total Failure	468
15.3	Summary and Conclusion	469
15.4	Related Reading	470
 Part III Applications of Reliability Techniques		
16	Retrofitting Reliability into Complex Systems	473
16.1	Wrappers and Toolkits	474
16.1.1	Wrapper Technologies	476
16.1.2	Introducing Robustness in Wrapped Applications	483
16.1.3	Toolkit Technologies	486
16.1.4	Distributed Programming Languages	488
16.2	Wrapping a Simple RPC Server	489
16.3	Wrapping a Web Site	491
16.4	Hardening Other Aspects of the Web	492
16.5	Unbreakable Stream Connections	496
16.5.1	Discussion	498
16.6	Reliable Distributed Shared Memory	498
16.6.1	The Shared Memory Wrapper Abstraction	499
16.6.2	Memory Coherency Options for Distributed Shared Memory	501

16.6.3	False Sharing	504
16.6.4	Demand Paging and Intelligent Prefetching	505
16.6.5	Fault Tolerance Issues	506
16.6.6	Security and Protection Considerations	506
16.6.7	Summary and Discussion	507
16.7	Related Reading	508
17	Software Architectures for Group Communication	509
17.1	Architectural Considerations in Reliable Systems	510
17.2	Horus: A Flexible Group Communication System	512
17.2.1	A Layered Process Group Architecture	514
17.3	Protocol Stacks	517
17.4	Using Horus to Build a Publish-Subscribe Platform and a Robust Groupware Application	519
17.5	Using Electra to Harden CORBA Applications	522
17.6	Basic Performance of Horus	523
17.7	Masking the Overhead of Protocol Layering	526
17.7.1	Reducing Header Overhead	529
17.7.2	Eliminating Layered Protocol Processing Overhead . . .	530
17.7.3	Message Packing	531
17.7.4	Performance of Horus with the Protocol Accelerator . . .	532
17.8	Scalability	532
17.9	Performance and Scalability of the Spread Toolkit	535
17.10	Related Reading	538
 Part IV Related Technologies		
18	Security Options for Distributed Settings	543
18.1	Security Options for Distributed Settings	543
18.2	Perimeter Defense Technologies	548
18.3	Access Control Technologies	551
18.4	Authentication Schemes, Kerberos, and SSL	554
18.4.1	RSA and DES	555
18.4.2	Kerberos	557
18.4.3	ONC Security and NFS	560
18.4.4	SSL Security	561
18.5	Security Policy Languages	564
18.6	On-The-Fly Security	566
18.7	Availability and Security	567
18.8	Related Reading	569
19	Clock Synchronization and Synchronous Systems	571
19.1	Clock Synchronization	571
19.2	Timed-Asynchronous Protocols	576
19.3	Adapting Virtual Synchrony for Real-Time Settings	584
19.4	Related Reading	586

20	Transactional Systems	587
20.1	Review of the Transactional Model	587
20.2	Implementation of a Transactional Storage System	589
20.2.1	Write-Ahead Logging	589
20.2.2	Persistent Data Seen Through an Updates List	590
20.2.3	Nondistributed Commit Actions	591
20.3	Distributed Transactions and Multiphase Commit	592
20.4	Transactions on Replicated Data	593
20.5	Nested Transactions	594
20.5.1	Comments on the Nested Transaction Model	596
20.6	Weak Consistency Models	599
20.6.1	Epsilon Serializability	600
20.6.2	Weak and Strong Consistency in Partitioned Database Systems	600
20.6.3	Transactions on Multidatabase Systems	602
20.6.4	Linearizability	602
20.6.5	Transactions in Real-Time Systems	603
20.7	Advanced Replication Techniques	603
20.8	Snapshot Isolation	606
20.9	Related Reading	607
21	Peer-to-Peer Systems and Probabilistic Protocols	609
21.1	Bimodal Multicast Protocol	609
21.1.1	Bimodal Multicast	612
21.1.2	Unordered ProbabilisticSend Protocol	614
21.1.3	Weakening the Membership Tracking Rule	616
21.1.4	Adding CASD-Style Temporal Properties and Total Ordering	617
21.1.5	Scalable Virtual Synchrony Layered over ProbabilisticSend	617
21.1.6	Probabilistic Reliability and the Bimodal Delivery Distribution	618
21.1.7	Evaluation and Scalability	621
21.1.8	Experimental Results	622
21.2	Astrolabe	623
21.2.1	How It Works	625
21.2.2	Peer-to-Peer Data Fusion and Data Mining	629
21.3	Other Applications of Peer-to-Peer Protocols	632
21.4	Related Reading	634
22	Appendix A: Virtually Synchronous Methodology for Building Dynamic Reliable Services	635
22.1	Introduction	636
22.2	Liveness Model	640
22.3	The Dynamic Reliable Multicast Problem	642
22.4	Fault-Recovery Multicast	646

22.4.1	Fault-Recovery Add/Get Implementation	646
22.4.2	Reconfiguration Protocol	646
22.5	Fault-Masking Multicast	648
22.5.1	Majorities-Based Tolerant Add/Get Implementation . . .	649
22.5.2	Reconfiguration Protocol for Majorities-Based Multicast	650
22.5.3	Reconfiguration Agreement Protocol	650
22.6	Coordinated State Transfer: The Virtual Synchrony Property . . .	653
22.7	Dynamic State Machine Replication and Virtually Synchronous Paxos	654
22.7.1	On Paxos Anomalies	655
22.7.2	Virtually Synchronous SMR	658
22.8	Dynamic Read/Write Storage	662
22.9	DSR in Perspective	662
22.9.1	Speculative-Views	664
22.9.2	Dynamic-Quorums and Cascading Changes	665
22.9.3	Off-line Versus On-line Reconfiguration	666
22.9.4	Paxos Anomaly	667
22.10	Correctness	667
22.10.1	Correctness of Fault-Recovery Reliable Multicast Solution	667
22.10.2	Correctness of Fault-Masking Reliable Multicast Solution	669
22.11	Further Readings	671
23	Appendix B: Isis² API	673
23.1	Basic Data Types	675
23.2	Basic System Calls	675
23.3	Timeouts	678
23.4	Large Groups	678
23.5	Threads	679
23.6	Debugging	679
24	Appendix C: Problems	681
	References	703
	Index	723



<http://www.springer.com/978-1-4471-2415-3>

Guide to Reliable Distributed Systems
Building High-Assurance Applications and Cloud-Hosted
Services

Birman, K.P.

2012, XXII, 730 p., Hardcover

ISBN: 978-1-4471-2415-3