

Table of Contents

Preface	1
Chapter 1: Introduction to NoSQL Databases	7
Consistency versus availability	8
ACID guarantees	9
Hash versus range partition	10
In-place updates versus appends	10
Row versus column versus column-family storage models	11
Strongly versus loosely enforced schemas	12
Summary	12
Chapter 2: MongoDB	13
Installing of MongoDB	14
MongoDB data types	16
The MongoDB database	17
MongoDB collections	18
MongoDB documents	19
The create operation	22
The read operation	24
Applying filters on fields	24
Applying conditional and logical operators on the filter parameter	25
The update operation	29
The delete operation	31
Data models in MongoDB	32
The references document data model	32
The embedded data model	33
Introduction to MongoDB indexing	35
The default <code>_id</code> index	35
Replication	39
Replication in MongoDB	39
Automatic failover in replication	40
Read operations	40
Sharding	40
Sharded clusters	41
Advantages of sharding	41
Storing large data in MongoDB	42
Summary	42
Chapter 3: Neo4j	43
What is Neo4j?	44

How does Neo4j work?	44
Features of Neo4j	45
Clustering	46
Neo4j Browser	46
Cache sharding	46
Help for beginners	47
Evaluating your use case	47
Social networks	47
Matchmaking	48
Network management	48
Analytics	48
Recommendation engines	49
Neo4j anti-patterns	49
Applying relational modeling techniques in Neo4j	50
Using Neo4j for the first time on something mission-critical	50
Storing entities and relationships within entities	50
Improper use of relationship types	51
Storing binary large object data	51
Indexing everything	51
Neo4j hardware selection, installation, and configuration	52
Random access memory	52
CPU	52
Disk	52
Operating system	53
Network/firewall	53
Installation	54
Installing JVM	54
Configuration	54
High-availability clustering	57
Causal clustering	57
Using Neo4j	58
Neo4j Browser	59
Cypher	60
Python	67
Java	69
Taking a backup with Neo4j	73
Backup/restore with Neo4j Enterprise	73
Backup/restore with Neo4j Community	75
Differences between the Neo4j Community and Enterprise Editions	77
Tips for success	78
Summary	79
References	79
Chapter 4: Redis	81
Introduction to Redis	82

What are the key features of Redis?	83
Performance	83
Tunable data durability	84
Publish/Subscribe	86
Useful data types	86
Expiring data over time	86
Counters	87
Server-side Lua scripting	87
Appropriate use cases for Redis	87
Data fits into RAM	87
Data durability is not a concern	87
Data at scale	88
Simple data model	88
Features of Redis matching part of your use case	88
Data modeling and application design with Redis	89
Taking advantage of Redis' data structures	89
Queues	89
Sets	89
Notifications	90
Counters	90
Caching	90
Redis anti-patterns	90
Dataset cannot fit into RAM	91
Modeling relational data	91
Improper connection management	91
Security	92
Using the KEYS command	92
Unnecessary trips over the network	92
Not disabling THP	92
Redis setup, installation, and configuration	93
Virtualization versus on-the-metal	93
RAM	94
CPU	95
Disk	95
Operating system	95
Network/firewall	95
Installation	96
Configuration files	97
Using Redis	100
redis-cli	100
Lua	104
Python	107
Java	113
Taking a backup with Redis	118
Restoring from a backup	119

Tips for success	121
Summary	121
References	122
Chapter 5: Cassandra	123
Introduction to Cassandra	124
What problems does Cassandra solve?	124
What are the key features of Cassandra?	125
No single point of failure	125
Tunable consistency	126
Data center awareness	126
Linear scalability	126
Built on the JVM	126
Appropriate use cases for Cassandra	126
Overview of the internals	127
Data modeling in Cassandra	130
Partition keys	130
Clustering keys	130
Putting it all together	131
Optimal use cases	132
Cassandra anti-patterns	133
Frequently updated data	134
Frequently deleted data	134
Queues or queue-like data	135
Solutions requiring query flexibility	135
Solutions requiring full table scans	135
Incorrect use of BATCH statements	136
Using Byte Ordered Partitioner	136
Using a load balancer in front of Cassandra nodes	136
Using a framework driver	136
Cassandra hardware selection, installation, and configuration	137
RAM	137
CPU	138
Disk	138
Operating system	138
Network/firewall	139
Installation using apt-get	139
Tarball installation	140
JVM installation	141
Node configuration	141
Running Cassandra	144
Adding a new node to the cluster	145
Using Cassandra	146
Nodetool	147
CQLSH	150

Python	153
Java	156
Taking a backup with Cassandra	160
Restoring from a snapshot	161
Tips for success	163
Run Cassandra on Linux	163
Open ports 7199, 7000, 7001, and 9042	164
Enable security	164
Use solid state drives (SSDs) if possible	164
Configure only one or two seed nodes per data center	164
Schedule weekly repairs	164
Do not force a major compaction	164
Remember that every mutation is a write	165
The data model is key	165
Consider a support contract	165
Cassandra is not a general purpose database	165
Summary	166
References	166
Chapter 6: HBase	169
Architecture	170
Components in the HBase stack	171
Zookeeper	171
HDFS	172
HBase master	173
HBase RegionServers	174
Reads and writes	175
The HBase write path	175
HBase writes – design motivation	177
The HBase read path	178
HBase compactions	179
System trade-offs	180
Logical and physical data models	181
Interacting with HBase – the HBase shell	184
Interacting with HBase – the HBase Client API	190
Interacting with secure HBase clusters	192
Advanced topics	193
HBase high availability	193
Replicated reads	194
HBase in multiple regions	195
HBase coprocessors	197
SQL over HBase	198
Summary	198
Chapter 7: DynamoDB	199
The difference between SQL and DynamoDB	200

Setting up DynamoDB	202
Setting up locally	202
Setting up using AWS	204
The difference between downloadable DynamoDB and DynamoDB web services	207
DynamoDB data types and terminology	208
Tables, items, and attributes	208
Primary key	210
Secondary indexes	212
Streams	214
Queries	217
Scan	218
Data types	219
Data models and CRUD operations in DynamoDB	220
Limitations of DynamoDB	232
Best practices	234
Summary	235
Chapter 8: InfluxDB	237
Introduction to InfluxDB	237
Key concepts and terms of InfluxDB	239
Data model and storage engine	242
Storage engine	243
Installation and configuration	245
Installing InfluxDB	245
Configuring InfluxDB	245
Production deployment considerations	247
Query language and API	248
Query language	248
Query pagination	260
Query performance optimizations	261
Interaction via Rest API	262
InfluxDB API client	267
InfluxDB with Java client	267
InfluxDB with a Python client	269
InfluxDB with Go client	270
InfluxDB ecosystem	273
Telegraf	273
Telegraf data management	274
Kapacitor	275
InfluxDB operations	279
Backup and restore	279
Backups	279
Restore	280
Clustering and HA	280

Retention policy	280
Monitoring	281
Summary	281
Other Books You May Enjoy	283
Index	287
