brief contents

Part 1	BACKGROUND AND FUNDAMENTALS					
	1 Hadoop in a heartbeat 3					
	2 Introduction to YARN 22					
Part 2	DATA LOGISTICS					
	3 Data serialization—working with text and beyond 61					
	4 • Organizing and optimizing data in HDFS 139					
	5 5 Moving data into and out of Hadoop 174					
Part 3	BIG DATA PATTERNS					
	6 Applying MapReduce patterns to big data 255					
	7 Utilizing data structures and algorithms at scale 302					
	8 u Tuning, debugging, and testing 337					
Part 4	BEYOND MAPREDUCE					
	9 SQL on Hadoop 387					
	10 - Writing a VADN application 495					

contents

preface xv acknowledgments xvii about this book xviii about the cover illustration xxiii

PART 1 BACKGROUND AND FUNDAMENTALS......1

Hadoop in a heartbeat 3

1.1 What is Hadoop? 4

Core Hadoop components 5 • The Hadoop ecosystem 10 Hardware requirements 11 • Hadoop distributions 12 • Who's using Hadoop? 14 • Hadoop limitations 15

- 1.2 Getting your hands dirty with MapReduce 17
- 1.3 Summary 21

7 Introduction to YARN 22

2.1 YARN overview 23

Why YARN? 24 • YARN concepts and components 26 YARN configuration 29

TECHNIQUE 1 Determining the configuration of your cluster 29 Interacting with YARN 31

TECHNIQUE 2 Running a command on your YARN cluster 31 TECHNIQUE 3 Accessing container logs 32 4 Aggregating container log files 36 TECHNIQUE YARN challenges 39 2.2 YARN and MapReduce 40 Dissecting a YARN MapReduce application 40 • Configuration 42 Backward compatibility 46 5 Writing code that works on Hadoop versions 1 TECHNIQUE and 2 47 48 Running a job 6 Using the command line to run a job TECHNIQUE 49 Monitoring running jobs and viewing archived jobs - 49 Uber jobs 50 TECHNIQUE 7 Running small MapReduce jobs 50 2.3YARN applications 52 NoSQL 53 • Interactive SQL 54 • Graph processing 54 Real-time data processing 55 - Bulk synchronous parallel 55 MPI 56 • In-memory 56 • DAG execution 56 2.4Summary 57 PART 2 DATA LOGISTICS.....59 Data serialization—working with text and beyond 61 3.1 Understanding inputs and outputs in MapReduce 62 Data input 63 • Data output 66 3.2Processing common serialization formats 68 XML 69 TECHNIQUE 8 MapReduce and XML 69 JSON 72 TECHNIQUE 9 MapReduce and JSON 73 3.3

Big data serialization formats 76 Comparing SequenceFile, Protocol Buffers, Thrift, and Avro 76

SequenceFile 78

TECHNIQUE10Working with SequenceFiles80TECHNIQUE11Using SequenceFiles to encode Protocol
Buffers87Protocol Buffers91 • Thrift92 • Avro93

TECHNIQUE 12 Avro's schema and code generation 93

Technique	13	Selecting the appropriate way to use Avro in				
		MapReduce 98				
TECHNIQUE	14	Mixing Avro and non-Avro data in MapReduce 99				
TECHNIQUE	15	Using Avro records in MapReduce 102				
TECHNIQUE	16	Using Avro key/value pairs in MapReduce 104				
TECHNIQUE	17	Controlling how sorting works in				
		MapReduce 108				
TECHNIQUE	18	Avro and Hive 108				
TECHNIQUE	19	Avro and Pig 111				

3.4 Columnar storage 113

Understanding object models and storage formats 115 • Parquet and the Hadoop ecosystem 116 • Parquet block and page sizes 117

- TECHNIQUE20Reading Parquet files via the command
lineTECHNIQUE21Reading and writing Avro data in Parquet with
JavaTECHNIQUE22Parquet and MapReduceTECHNIQUE23Parquet and Hive/ImpalaTECHNIQUE23Parquet and Hive/Impala
- TECHNIQUE 24 Pushdown predicates and projection with Parquet 126

Parquet limitations 128

3.5 Custom file formats 129

Input and output formats 129

TECHNIQUE 25 Writing input and output formats for CSV 129 The importance of output committing 137

3.6 Chapter summary 138

A Organizing and optimizing data in HDFS 139

4.1 Data organization 140 Directory and file layout 140 • Data tiers 141 • Partitioning 142 TECHNIQUE 26 Using MultipleOutputs to partition your data 142 27 Using a custom MapReduce partitioner 145 TECHNIQUE Compacting 148 TECHNIQUE 28 Using filecrush to compact data 149 TECHNIQUE 29 Using Avro to store multiple small binary files 151 Atomic data movement 157 4.2 Efficient storage with compression 158 Picking the right compression codec for your Technique 30

data 159

Technique	31	Compression with HDFS, MapReduce, Pig,
		and Hive 163
Technique	32	Splittable LZOP with MapReduce, Hive, and
		Pig 168

4.3 Chapter summary 173

Moving data into and out of Hadoop 174

- 5.1 Key elements of data movement 175
- 5.2 Moving data into Hadoop 177

Roll your own ingest 177

	Technique	33	Using the CLI to load files 178					
	Technique	34	Using REST to load files 180					
	Technique	35	Accessing HDFS from behind a firewall 183					
	TECHNIQUE	36	Mounting Hadoop with NFS 186					
	TECHNIQUE	37	Using DistCp to copy data within and between					
			clusters 188					
	Technique	38	Using Java to load files 194					
	Continuo	us mo	vement of log and binary files into HDFS 196					
	TECHNIQUE	39	Pushing system log messages into HDFS with Flume 197					
	Technique	40	An automated mechanism to copy files into HDFS 204					
	Technique	41	Scheduling regular ingress activities with Oozie 209					
	Databases	Databases 214						
	TECHNIQUE <i>HBase</i>	42 227	Using Sqoop to import data from MySQL 215					
	Technioue	43	HBase ingress into HDFS 227					
	TECHNIQUE	44	MapReduce with HBase as a data source 230					
	Importing	g data	from Kafka 232					
5.3	Moving da	Moving data into Hadoop 234						
	TECHNIQUE	45	Using Camus to copy Avro data from Kafka into HDFS 234					
5.4	Moving da	Moving data out of Hadoop 241						
	Roll your	own e	gress 241					
	Technique	46	Using the CLI to extract files 241					
	TECHNIQUE	47	Using REST to extract files 242					
	TECHNIQUE	48	Reading from HDFS when behind a					
			firewall 243					
	Technique	49	Mounting Hadoop with NFS 243					
	Technique	50	Using DistCp to copy data out of Hadoop 244					

		TECHNIQUE Automate	51 d file e	Using Java to extract files 245 gress 246
		TECHNIQUE	52	An automated mechanism to export files from HDFS 246
		Databases	s 242	7
		Technique <i>NoSQL</i>	53 251	Using Sqoop to export data to MySQL 247
	5.5	Chapter su	ımma	ary 252
Part 3	B IG I	DATA PAT	ΓERN	vs253
6	Apply	ing MapRe	educe	e patterns to big data 255
U	6.1	Joining 2	256	
		Technique Technique <i>Map-side</i>	54 55 joins	Picking the best join strategy for your data 257 Filters, projections, and pushdowns 259 <i>260</i>
		TECHNIQUE	56	Joining data where one dataset can fit into
		Technique Technique	57 58	Performing a semi-join on large datasets 264 Joining on presorted and prepartitioned
		Reduce-si	de join	s 271
		TECHNIQUE	59	A basic repartition join 271
		TECHNIQUE	60	Optimizing the repartition join 275
		I ECHNIQUE	61	data 279
		Data skeu	v in ree	duce-side joins 283
		TECHNIQUE	62	Joining large datasets with high join-key cardinality 284
		TECHNIQUE	63	Handling skews generated by the hash partitioner 286
	6.2	Sorting 2	287	1
		Secondary	sort	288
		TECHNIQUE Total ord	64 er sorti	Implementing a secondary sort 289 ing 294
		TECHNIQUE	65	Sorting keys across multiple reducers 294
	6.3	Sampling	297	
		TECHNIQUE	66	Writing a reservoir-sampling InputFormat 297
	6.4	Chapter su	ımma	ary 301

7 Utilizing data structures and algorithms at scale 302

7.1Modeling data and solving problems with graphs 303 Modeling graphs 304 • Shortest-path algorithm 304 TECHNIQUE 67 Find the shortest distance between two users 305Friends-of-friends algorithm 313 TECHNIQUE 68 Calculating FoFs 313 Using Giraph to calculate PageRank over a web graph 319 7.2Modeling data and solving problems with graphs 321 Technique 69 Calculate PageRank over a web graph 322 7.3 Bloom filters 326 TECHNIQUE 70 Parallelized Bloom filter creation in MapReduce 328 7.4HyperLogLog 333 A brief introduction to HyperLogLog 333 71Using HyperLogLog to calculate unique TECHNIQUE counts 335 7.5Chapter summary 336 Tuning, debugging, and testing 337 8.1 Measure, measure, measure 338 8.2 Tuning MapReduce 339 *Common inefficiencies in MapReduce jobs* 339 Technique 72 Viewing job statistics 340 Map optimizations 343 TECHNIQUE 73Data locality 343 TECHNIQUE 74Dealing with a large number of input splits 344 75TECHNIQUE Generating input splits in the cluster with YARN 346 Shuffle optimizations 347 TECHNIQUE 76 Using the combiner 347 77 TECHNIQUE Blazingly fast sorting with binary comparators 349 TECHNIQUE 78 Tuning the shuffle internals 353 Reducer optimizations 356 TECHNIQUE 79 Too few or too many reducers 356 General tuning tips 357

	TECHNIQUE	80	Using stack dumps to discover upoptimized							
	TEOHNQUE	00	user code 358							
	TECHNIQUE	81	Profiling your map and reduce tasks 360							
8.3	Debugging	g 36	52							
	Accessing	Accessing container log output 362								
	TECHNIQUE	82	Examining task logs 362							
	Accessing	conta	iner start scripts 363							
	TECHNIQUE	83	Figuring out the container startup command 363							
	Debuggin	Debugging OutOfMemory errors 365								
	TECHNIQUE	84	Force container JVMs to generate a heap dump 365							
	MapRedu	ice coa	ling guidelines for effective debugging 365							
	TECHNIQUE	85	Augmenting MapReduce code for better de bugging 365							
8.4	Testing Ma	apRe	duce jobs 368							
	Essential	ingred	lients for effective unit testing 368 • MRUnit 370							
	Technique <i>LocalJobR</i>	86 Runne	Using MRUnit to unit-test MapReduce 371 r 378							
	TECHNIQUE	87	Heavyweight job testing with the LocalJobRunner 378							
	MiniMR	YarnC	lluster 381							
	Technique Integratio	88 n and	Using MiniMRYarnCluster to test your jobs 381 <i>QA testing 382</i>							
8.5	5 Chapter summary 383									
BEYC	ond Mapl	Red	UCE							
SQL a	on Hadoop	38	37							
9.1	Hive 388									
	Hive basi	cs 3	88 • Reading and writing data 391							
	TECHNIQUE	89	Working with text files 391							
	Technique User-defin	90 ved fu	Exporting data to local disk 395 nctions in Hive 396							
	Technique <i>Hive perfe</i>	91 orman	Writing UDFs 396 ace 399							
	TECHNIQUE	92	Partitioning 399							
	TECHNIQUE	93	Tuning Hive joins 404							

PART 4

(

xiii

9.2 Impala 409

Impala vs. Hive 410 • Impala basics 410

TECHNIQUE94Working with text410TECHNIQUE95Working with Parquet412TECHNIQUE96Refreshing metadata413User-defined functions in Impala414

TECHNIQUE 97 Executing Hive UDFs in Impala 415

9.3 Spark SQL 416

Spark 101 417 • Spark on Hadoop 419 • SQL with Spark 419

TECHNIQUE98Calculating stock averages with Spark SQL420TECHNIQUE99Language-integrated queries422TECHNIQUE100Hive and Spark SQL423

9.4 Chapter summary 423

Writing a YARN application 425

10.1	Fundamentals of building a YARN application					
	Actors	426 • The mechanics of a YARN application	427			

- 10.2 Building a YARN application to collect cluster statistics 429
 - TECHNIQUE 101 A bare-bones YARN client 429
 - TECHNIQUE 102 A bare-bones ApplicationMaster 434
 - TECHNIQUE 103 Running the application and accessing logs 438
 - TECHNIQUE 104 Debugging using an unmanaged application master 440

10.3 Additional YARN application capabilities 443

RPC between components 443 • Service discovery 444 Checkpointing application progress 444 • Avoiding split-brain 444 Long-running applications 444 • Security 445

- 10.4 YARN programming abstractions 445 *Twill* 446 • Spring 448 • REEF 450 • Picking a YARN *API abstraction* 450
- 10.5 Summary 450
- appendix Installing Hadoop and friends 451 index 475

bonus chapters available for download from www.manning.com/holmes2

chapter 11 Integrating R and Hadoop for statistics and morechapter 12 Predictive analytics with Mahout