

Table of Contents

Foreword.....	ix
Preface.....	xiii
1. What Is Apache Hudi?.....	1
The Evolution of Data Management Architectures	2
The Rise of Data Lakehouses	3
Uber’s “Transactional Data Lake” Problem	5
What Is Hudi?	8
The Hudi Stack	10
Native Table Format	11
Pluggable Table Format	12
Storage Engine	12
Programming API	14
User Access	16
Shared Platform Components	17
Hudi in the Real World	17
Summary	18
2. Getting Started with Hudi.....	21
Basic Operations	22
Create the Table	22
Insert, Update, Delete, and Fetch Records	26
Choose a Table Type	29
Create a Merge-on-Read Table	29
MOR Table’s Layout After Writes	30
Copy-on-Write Versus Merge-on-Read	35
Advanced Usage	39

Create Table As Select	39
Merge Source Data into the Table	40
Update and Delete Using Nonrecord Key Fields	41
Time Travel Query	42
Incremental Query	42
Summary	43
3. Writing to Hudi.....	45
Breaking Down the Write Flow	46
Start Commit	47
Prepare Records	47
Partition Records	49
Write to Storage	49
Commit Changes	49
Summarize the Upsert Flow.....	50
Exploring Write Operations	52
Define Table Properties.....	52
Use INSERT INTO	53
Perform Partial Merge with MERGE INTO	60
Perform Deletion	61
Overwrite Partition or Table	62
Highlighting Noteworthy Features	63
Key Generators	63
Merge Modes	66
Schema Evolution on Write	67
Bootstrapping	68
Summary	70
4. Reading from Hudi.....	73
Integrating with Query Engines	74
Query Lifecycle	74
Data Catalog	75
Hudi Integration	76
Exploring Query Types.....	79
Snapshot Query	80
Time Travel Query	81
Incremental Query: The Latest-State Mode	82
Incremental Query: The Change Data Capture Mode	84
Highlighting Noteworthy Features	87
Streaming Read	87
Schema Evolution on Read	88

Read Using Rust or Python	89
Summary	91
5. Achieving Efficiency with Indexing.....	93
Overview of the Indexes in Hudi	94
Index Acceleration for Writes	95
General-Purpose Multimodal Indexing	96
Writer-Side Indexes	99
Comparison of Writer Indexing Choices	107
Index Acceleration for Reads	109
Data Skipping	110
Equality Matching	113
Indexing on Expressions	114
Build the Right Indexes	115
Summary	116
6. Maintaining and Optimizing Hudi Tables.....	117
Table Service Overview	118
Deployment Mode: Inline	119
Deployment Mode: Async Execution	120
Deployment Mode: Standalone	121
Choosing a Suitable Mode	123
Compaction	124
Schedule Compaction	125
Execute Compaction	127
Clustering	128
Schedule Clustering	129
Execute Clustering	130
Layout Optimization Strategies	130
Clustering Versus Compaction	132
Cleaning	132
Schedule Cleaning	133
Execute Cleaning	134
Indexing	134
Summary	136
7. Concurrency Control in Hudi.....	139
Why Concurrency Control Is Harder in Data Lakehouses	139
Concurrency Control Techniques	141
Multiwriter Scenarios	142
Why Multiwriters Are Necessary	142

Multiwriter Scenarios for OCC	143
Multiwriter Scenarios for NBCC and MVCC	144
The Simple Default: Single Writer with Table Services	144
How Hudi Handles Concurrency Control	145
The Foundations of Hudi's Concurrency Control	145
The Three-Step Commit Process	148
Conflict Detection and Resolution	150
Locking Mechanisms	151
Challenges in Multiwriter Systems	153
Using Multiwriter Support in Hudi	153
Enabling Multiwriter Support	154
Configuring the Locking Mechanism	154
Multiwriters Using Hudi Streamer	155
Multiwriters Using Spark Data Source Writer	156
Single Writer and Multiple Table Services	157
Disabling Multiwriter Support	157
Tips and Best Practices	157
Implement Partitioning and File Grouping	158
Enable Early Conflict Detection	158
Optimize Locking Mechanisms	159
Run Asynchronous Table Services	160
Reduce Write Conflicts and Wasted Resources	160
Prevent Data Duplication When Using Multiple Writers	160
Summary	160
8. Building a Lakehouse Using Hudi Streamer	163
Alcubierre's Data Silo Woes	164
Data Quality Assurance and Deduplication	164
Heterogeneous Data and Schema Evolution	165
Data Management, Localization, and Consistency	166
Problem Recap	166
Lakehouse Architecture to the Rescue	167
What Is Hudi Streamer?	168
Getting Started with Hudi Streamer	169
Ingesting Data from S3	170
Ingesting Data from Kafka	170
Ingesting Data from RDBMS	172
Hudi Streamer in Action	173
Preparing the Upstream Source	175
Setting Up Hudi Streamer	178
Unlocking the Power of Analytics	183

Exploring the Hudi Streamer Options	185
General Options	186
Source Options	187
Operational Options	188
Summary	189
9. Running Hudi in Production.....	191
Operating with Ease	192
Getting to Know the CLI	192
Performing Table Operations	197
Integrating into the Platform	204
Triggering Post-Commit Callbacks	204
Wiring Up Monitoring Systems	209
Syncing with Catalogs	213
Performance Tuning	218
Storage Layout Tuning	218
Write Performance Tuning	219
Read Performance Tuning	221
Table Services Tuning	222
Summary	223
10. Building an End-to-End Lakehouse Solution.....	225
Architecture Overview	226
RetailMax Corp: A Real-World Lakehouse Scenario	227
Implementing Medallion Architecture with Hudi	229
Configuring RetailMax's Hudi Tables	229
Record Keys	230
Ordering Field	230
Partitioning	230
Table Types	231
Bronze Layer: Ingesting Upstream Data	232
Setting Up Upstream Data Sources	232
Streaming Mutable, Transactional Data with Debezium, Flink, and Hudi	233
Ingesting Application Event Streams with Hudi Kafka Connect Sink	235
Silver Layer: Creating Derived Datasets	237
Goals of the Silver Layer for RetailMax	238
Streaming-Based Transformations with Hudi Streamer	239
Batch and Incremental Transformations with Spark SQL	239
Maintaining Data Quality and Consistency in the Silver Layer	241
Gold Layer: Querying the Lakehouse for Insights	242
Interactive Analytics with Trino	243

188	Batch Analytics and Reporting with Spark SQL	243
188	Advanced Querying: Time Travel and Point-in-Time Analysis	244
187	Business Layer: AI-Driven Insights for RetailMax	245
188	Preparing Data for AI/Machine Learning in the Gold Layer	246
189	Building a Knowledge Base for LLM-Powered Applications with Ray and Hudi	247
191	Operationalizing and Optimizing the Hudi Lakehouse	249
192	Concurrency Control and Multiwriter Scenarios	250
192	Monitoring the Lakehouse	250
197	Data Resilience	251
199	Performance Benchmarks and Considerations	251
204	Summary	254
209	Multiwriters Using Hudi Streamer	251
217	Index	257
218	Single Writer and Multiple Table Services	218
218	Disabling Multiwriter Support	218
219	Best Practices	219
221	Improved Partitioning and File Grouping	221
222	Enable Early Conflict Detection	222
223	Optimize Locking Mechanisms	223
223	Use Asynchronous Table Services	223
225	Building an End-to-End Lakehouse Solution	225
226	Prevent Data Duplication When Using Multiple Writers	226
227	Summary	227
229	Implementing Medallion Architecture with Hudi	229
229	Configuring RetailMax's Hudi Tables	229
230	Subscriber Data Silo Woes	230
230	Data Quality Assurance and Deduplication	230
230	Heterogeneous Data and Schema Evolution	230
231	Data Management, Localization, and Consistency	231
232	Problem Key	232
232	Lakehouse Architecture to the Rescue	232
233	Setting Up Upstream Data Sources	233
233	Streaming Mutable, Transactional Data with Delta	233
235	Ingesting Application Event Streams with Hudi Kafka Eventstreams	235
237	ETL and Gateway	237
238	Goals of the Silver Layer for RetailMax	238
239	Streaming-Based Transformations with Hudi Structured Streaming	239
239	Batch and Incremental Transformations with Spark SQL	239
241	Maintaining Data Quality and Consistency in the Silver Layer	241
242	Gold Layer: Querying the Lakehouse for Insights	242
243	Interactive Analytics with Trino	243