SNIA DEVELOPER CONFERENCE

SDC 24

BY Developers FOR Developers

September 16-18, 2024
Santa Clara, CA

Open Standards for Open Lakehouses

Presentation Subtitle

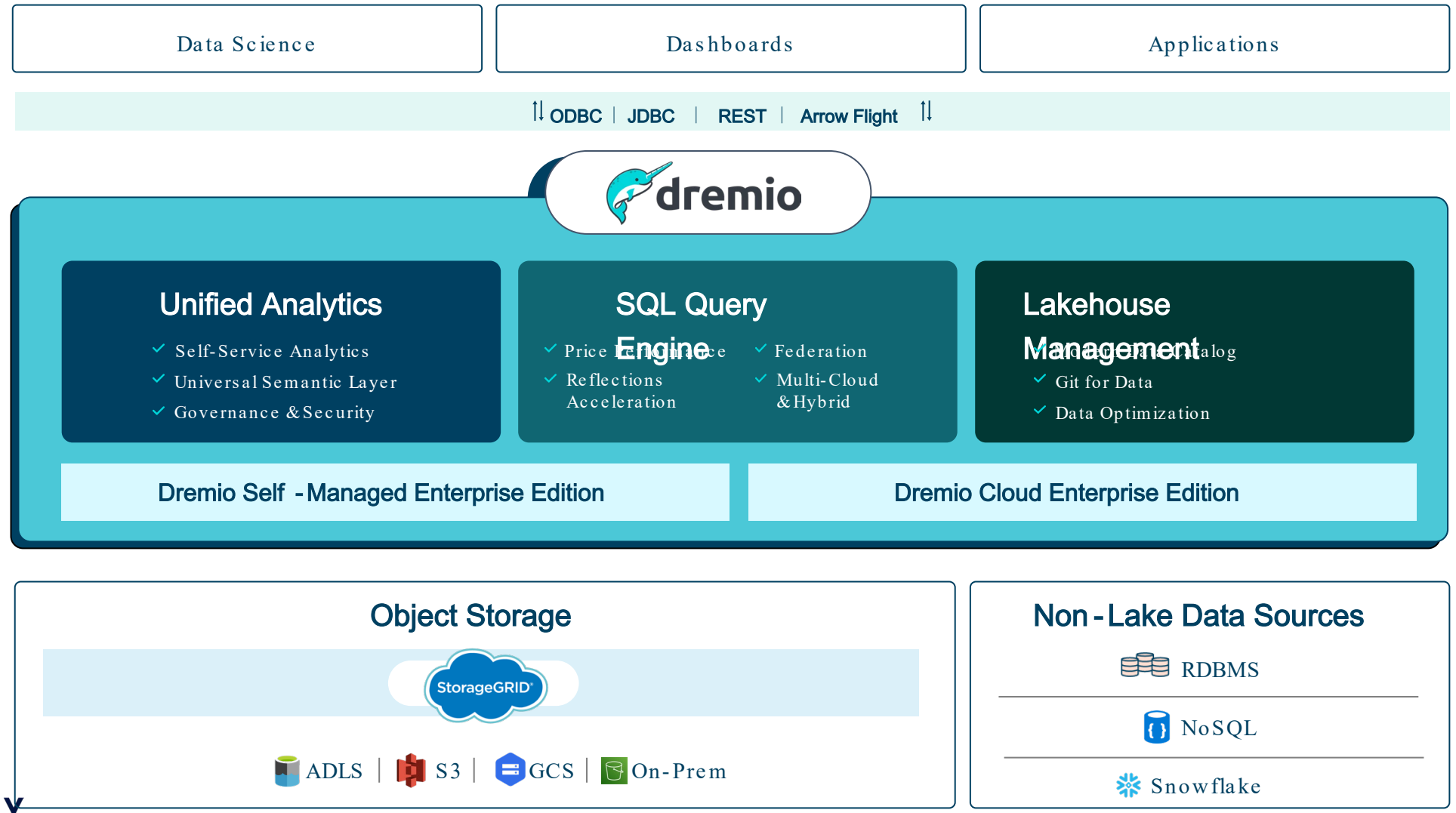
Presented by

Link to Download Slides

Who Am I?

And where am I from?

Dremio: The Unified Lakehouse Platform for Self-Service Analytics & AI



Unified Analytics

BI tools, data science notebooks, SQL editors

ODBC | JDBC | REST | Arrow Flight

UNIFIED ANALYTICS

Self-Service Analytics

- Shareable, governed data Views
- GenAI Text-to-SQL
- Comprehensive BI integration
- Embedded SQL runner

Semantic Layer

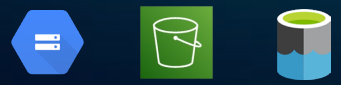
- Business-focused virtual data sets and marts
- Seamless search
- Intuitive, user-generated project Wikis

Governance & Security

- Role-Based Access (RBAC)
- Fine-grained access control
- Auditing & Query history
- Data Lineage
- Identify Management & SSO integration

Iceberg, Delta Lake | Parquet, ORC, JSON, CSV Readers | ARP Connectors

Cloud Object Storage



On-Prem

SQL Query Engine

BI tools, data science notebooks, SQL editors

ODBC | JDBC | REST | Arrow Flight

SQL QUERY ENGINE

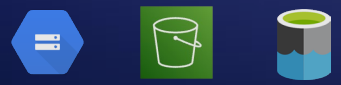
Price Performance	• Auto-scaling/elastic engines • Multi-engine architecture • Workload management, query routing
Reflections Query Acceleration	• Reflections Acceleration via optimized relational cache • Automated Reflections Recommender • Columnar Cloud Cache (C3)
Federation	• Query federation • Connector Ecosystem

Multi-Cloud & Hybrid

- On-premise
- In the cloud
- Multi-cloud
- Hybrid architecture

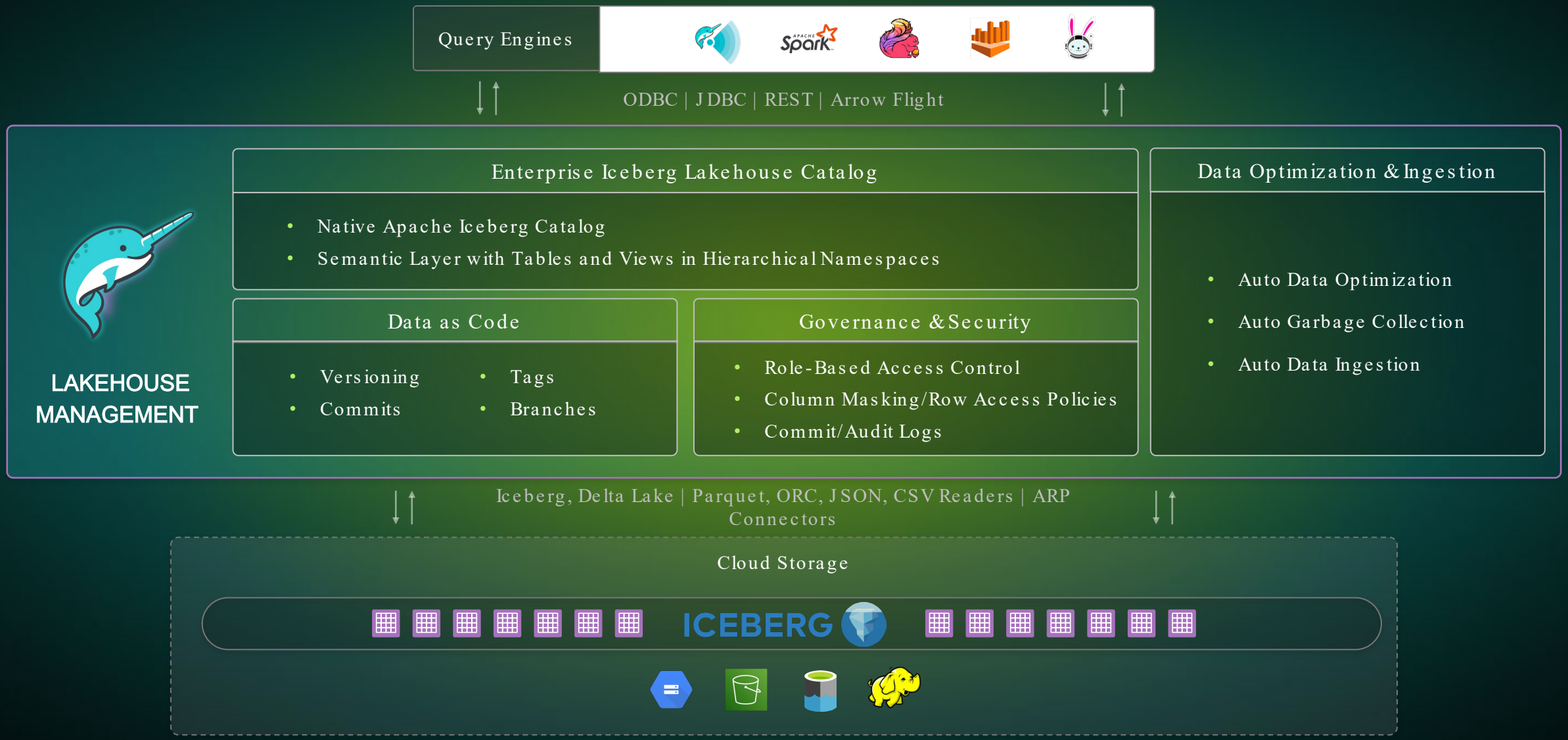
Iceberg, Delta Lake | Parquet, ORC, JSON, CSV Readers | ARP Connectors

Cloud Object Storage

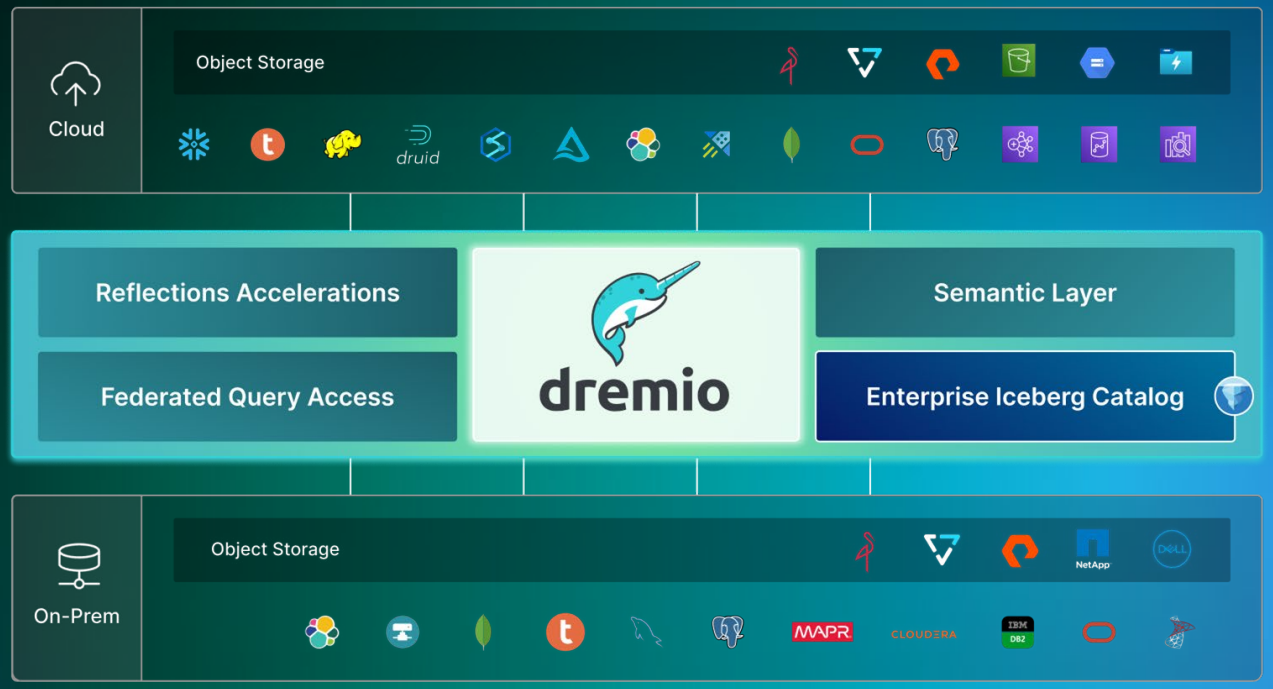


On-Prem

Lakehouse Management



Dremio Hybrid Iceberg Lakehouse

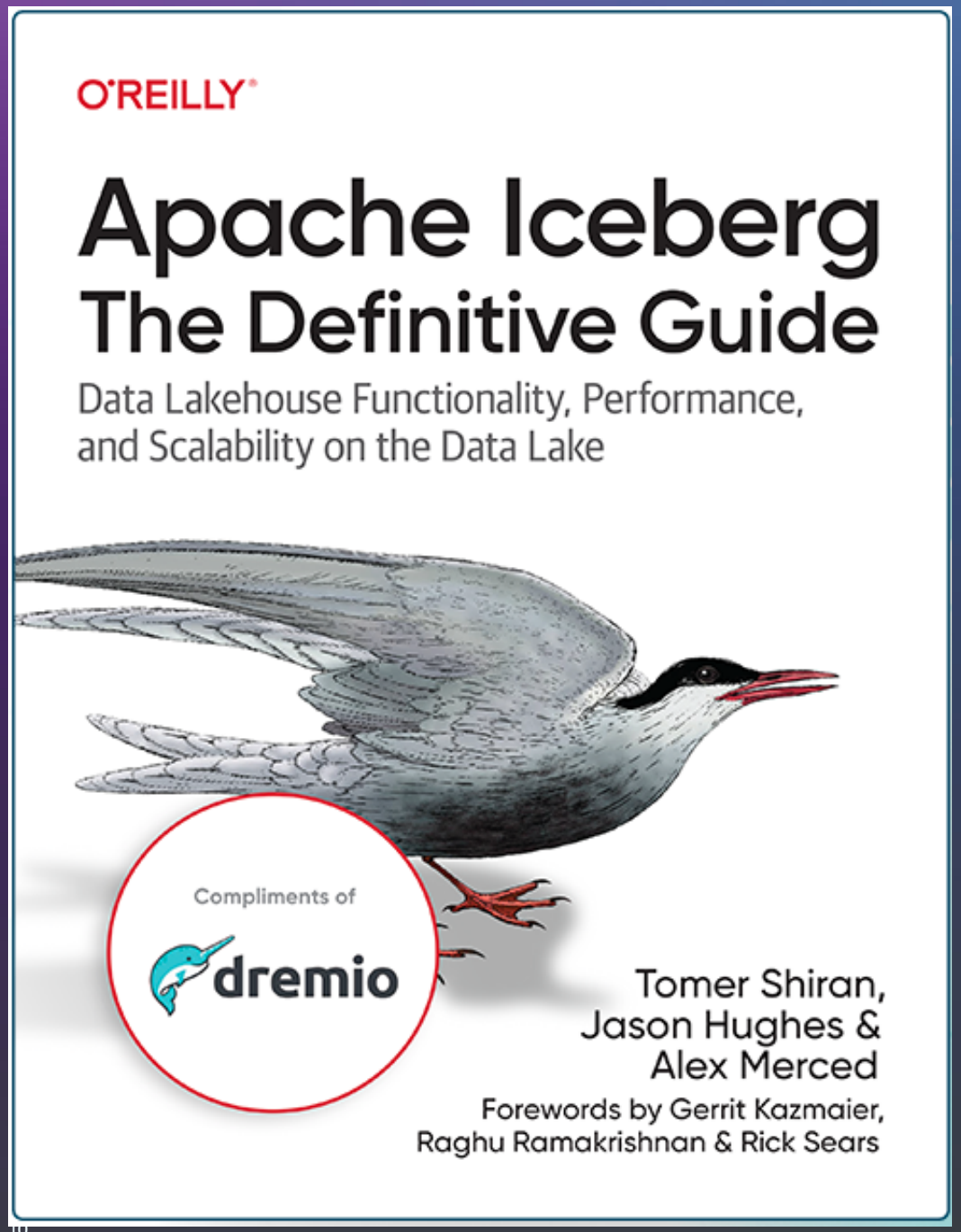


- Enterprise Hybrid Cloud Iceberg Catalog
- Leverage on-prem and cloud based storage
- Enable self service across all data sources

Alex Merced

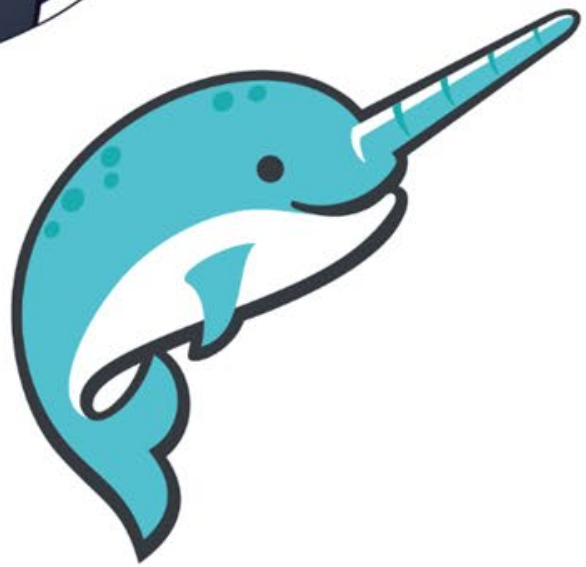
Senior Technical Evangelist, Dremio

Alex Merced is a senior technical evangelist at Dremio with experience as a developer and instructor. His professional journey includes roles at GenEd Systems, Crossfield Digital, CampusGuard, and General Assembly. He co-authored **"Apache Iceberg: The Definitive Guide"** published by O'Reilly and has spoken at notable events such as Data Day Texas and Data Council. Alex is passionate about technology, sharing his expertise through blogs, videos, podcasts like Datanation and Web Dev 101, and contributions to the JavaScript and Python communities with libraries like SencilloDB and CoquitoJS.

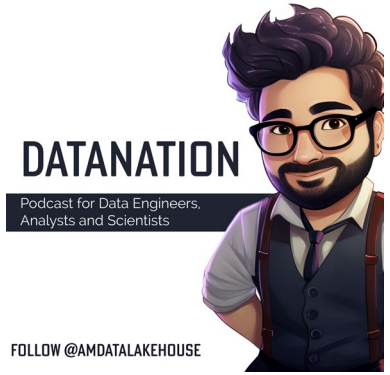

O'REILLY
Apache Iceberg
The Definitive Guide
Data Lakehouse Functionality, Performance, and Scalability on the Data Lake
Compliments of
dremio
Tomer Shiran, Jason Hughes & Alex Merced
Forewords by Gerrit Kazmaier, Raghu Ramakrishnan & Rick Sears

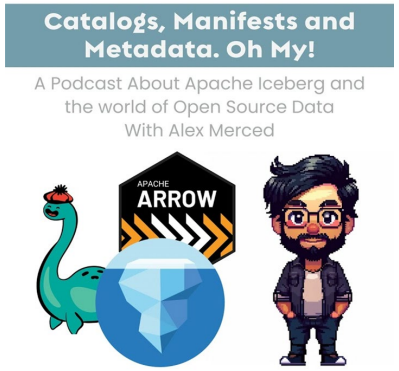
An Apache Iceberg Crash Course
10 Sessions

https://drmevn.fyi/sniadevcrashco



Podcasts





alexmerced.com/data

Link to Download Slides

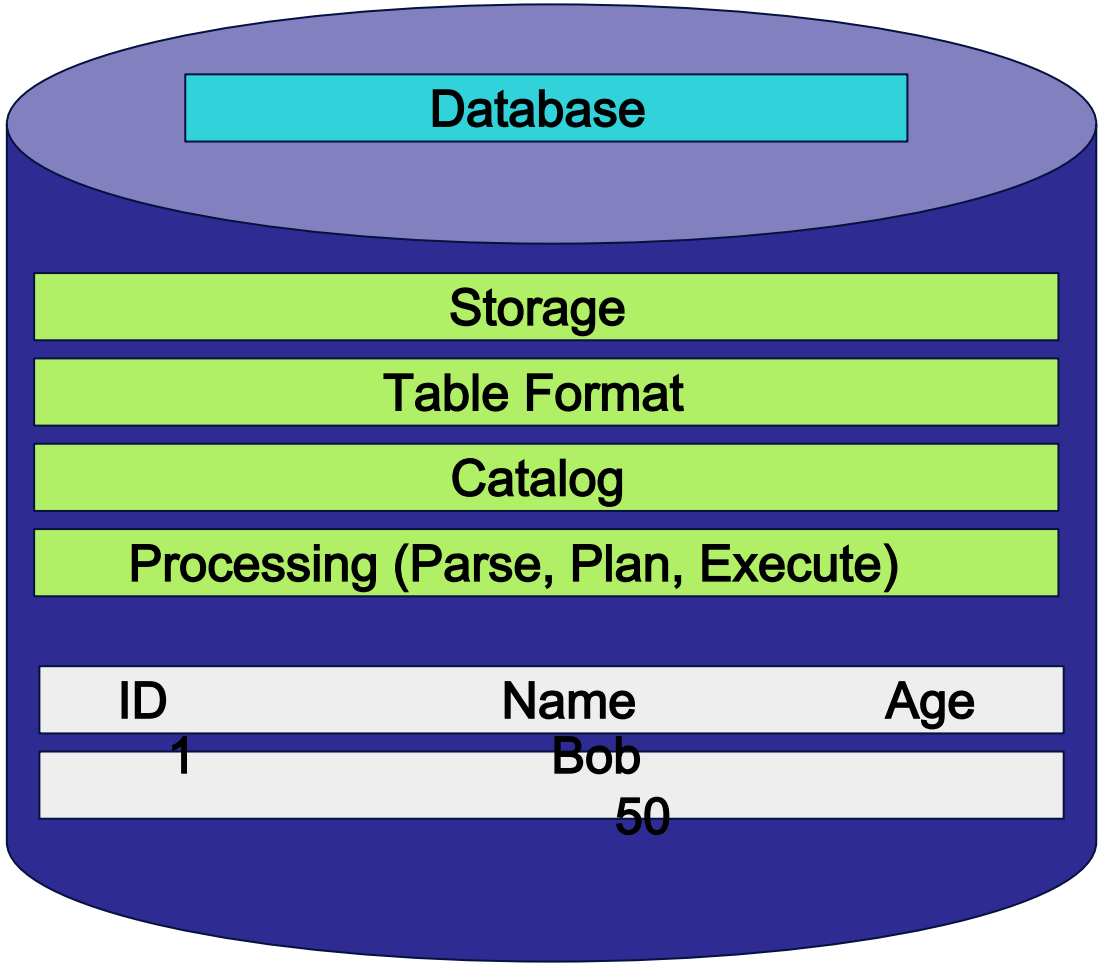
Agenda

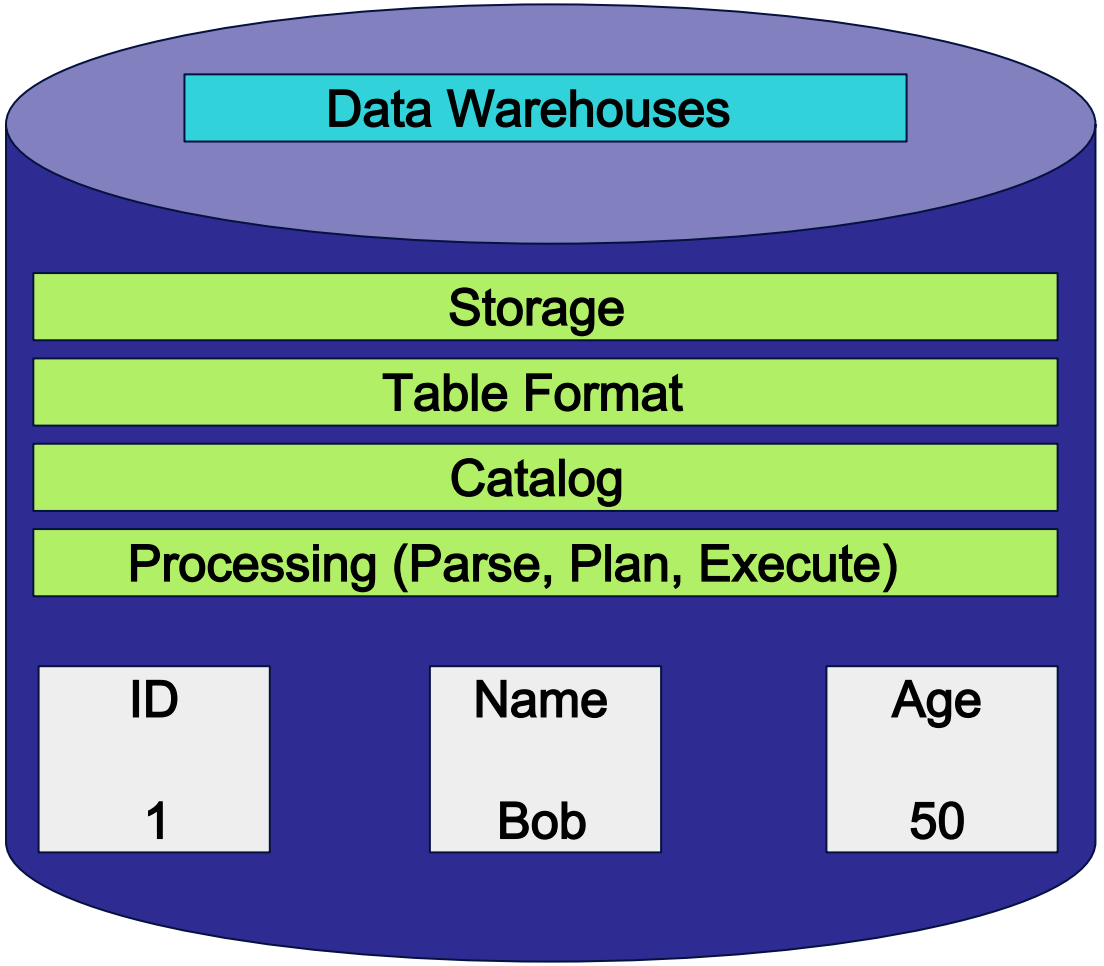
- What is a Data Lakehouse
- What is Apache Parquet
- What is Apache Iceberg
- What is Apache Arrow
- What is Nessie

What is a Data Lakehouse

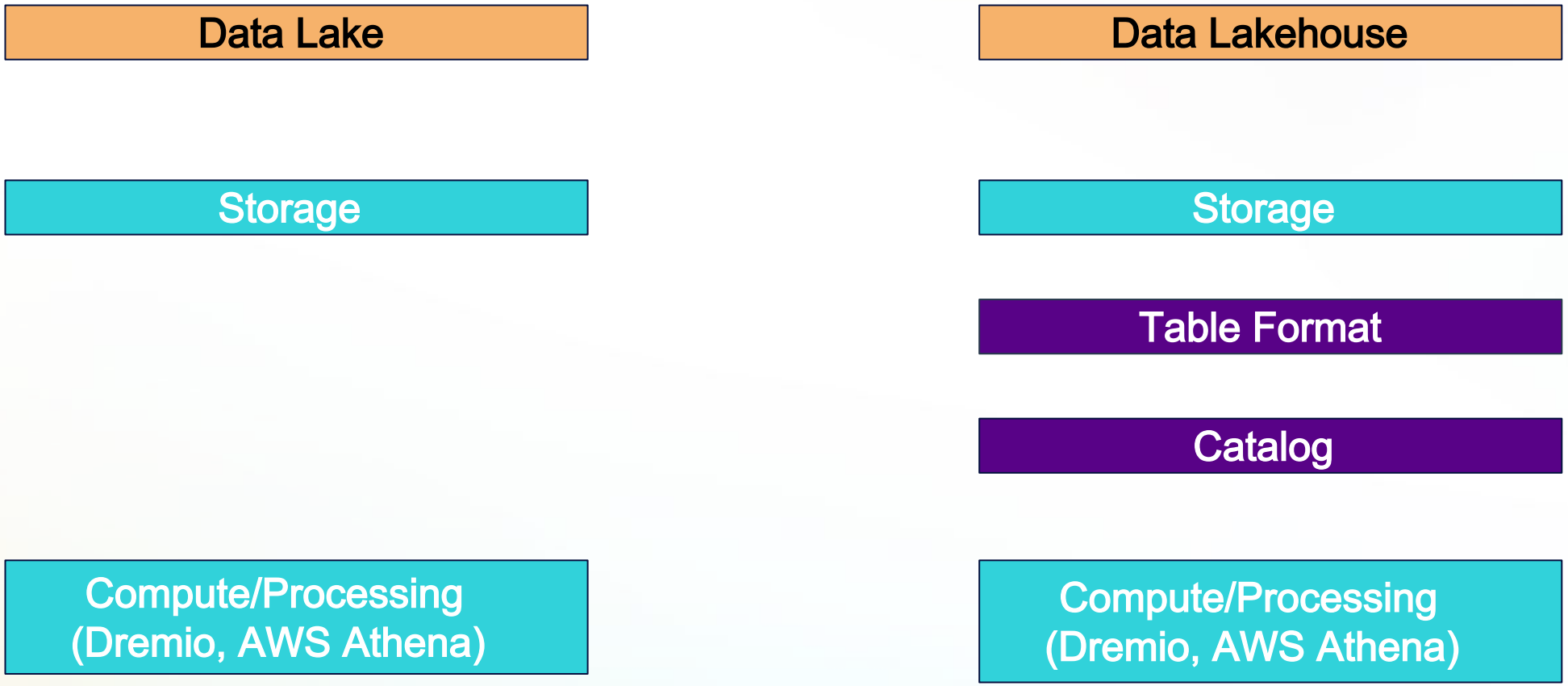
Turning the Data Lake into a Data Warehouse

Traditional Data Systems





What is a Data Lakehouse?



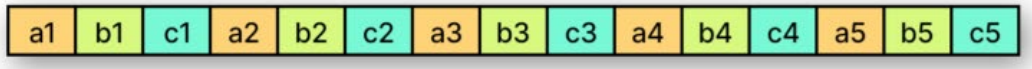
What is Apache Parquet

Binary Analytics Optimized File Format

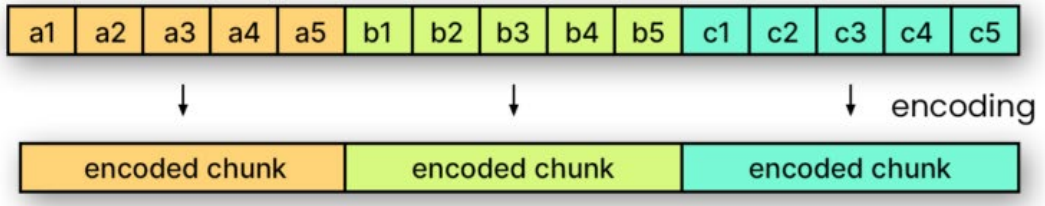
Logical table representation

a	b	c
a1	b1	c1
a2	b2	c2
a3	b3	c3
a4	b4	c4
a5	b5	c5

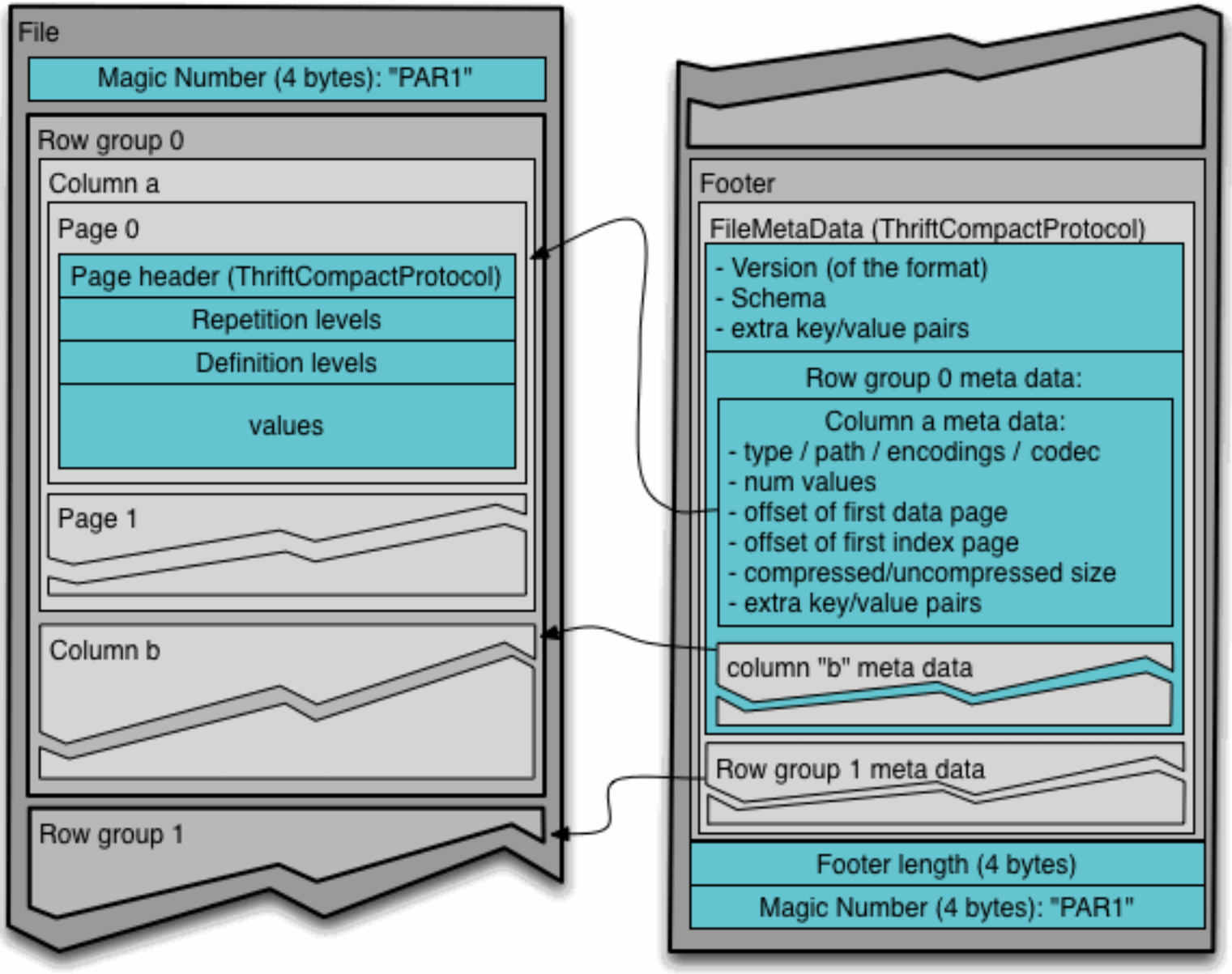
Row Layout



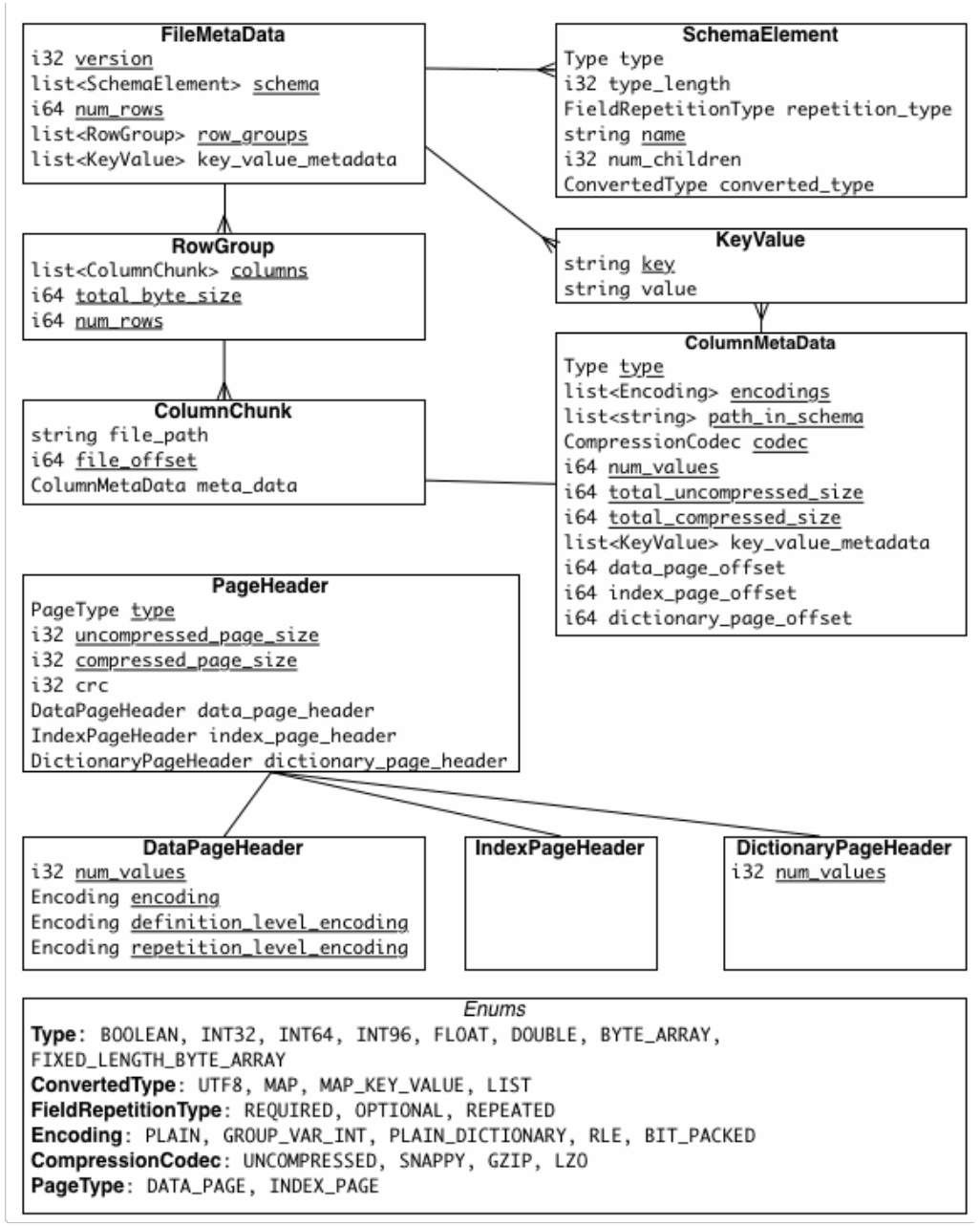
Column Layout



Parquet


File
Magic Number (4 bytes): "PAR1"
Row group 0
Column a
Page 0
Page header (ThriftCompactProtocol)
Repetition levels
Definition levels
values
Page 1
Column b
Row group 1
Footer
FileMetaData (ThriftCompactProtocol)
- Version (of the format)
- Schema
- extra key/value pairs
Row group 0 meta data:
Column a meta data:
- type / path / encodings / codec
- num values
- offset of first data page
- offset of first index page
- compressed/uncompressed size
- extra key/value pairs
column "b" meta data
Row group 1 meta data
Footer length (4 bytes)
Magic Number (4 bytes): "PAR1"

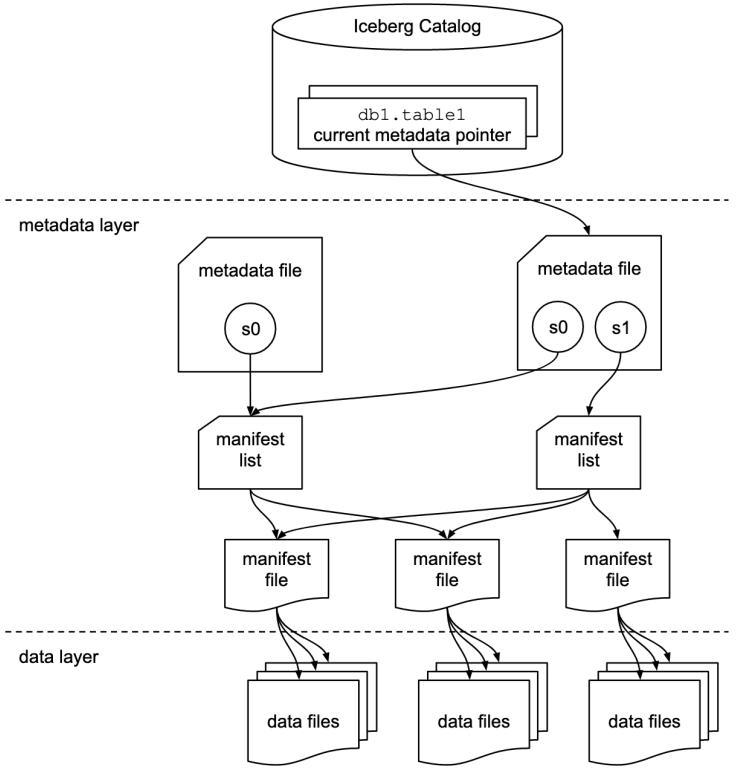

Parquet


FileMetaData
i32 version
list<SchemaElement> schema
i64 num_rows
list<RowGroup> row_groups
list<KeyValue> key_value_metadata
SchemaElement
Type type
i32 type_length
FieldRepetitionType repetition_type
string name
i32 num_children
ConvertedType converted_type
RowGroup
list<ColumnChunk> columns
i64 total_byte_size
i64 num_rows
KeyValue
string key
string value
ColumnChunk
string file_path
i64 file_offset
ColumnMetaData meta_data
ColumnMetaData
Type type
list<Encoding> encodings
list<string> path_in_schema
CompressionCodec codec
i64 num_values
i64 total_uncompressed_size
i64 total_compressed_size
list<KeyValue> key_value_metadata
i64 data_page_offset
i64 index_page_offset
i64 dictionary_page_offset
PageHeader
PageType type
i32 uncompressed_page_size
i32 compressed_page_size
i32 crc
DataPageHeader data_page_header
IndexPageHeader index_page_header
DictionaryPageHeader dictionary_page_header
DataPageHeader
i32 num_values
Encoding encoding
Encoding definition_level_encoding
Encoding repetition_level_encoding
IndexPageHeader
DictionaryPageHeader
i32 num_values
Enums
Type: BOOLEAN, INT32, INT64, INT96, FLOAT, DOUBLE, BYTE_ARRAY, FIXED_LENGTH_BYTE_ARRAY
ConvertedType: UTF8, MAP, MAP_KEY_VALUE, LIST
FieldRepetitionType: REQUIRED, OPTIONAL, REPEATED
Encoding: PLAIN, GROUP_VAR_INT, PLAIN_DICTIONARY, RLE, BIT_PACKED
CompressionCodec: UNCOMPRESSED, SNAPPY, GZIP, LZO
PageType: DATA_PAGE, INDEX_PAGE

What is Apache Iceberg

Open Table Format for Database Like Tables on The Lake

Apache Iceberg



Apache Iceberg's approach is to define the table through three layers of metadata. These categories are:

- **metadata files** that define the table
- **manifest lists** that define a snapshot of the table, with a list of manifests that make up the snapshot and metadata about their data
- **manifests** is a list of data files along with metadata on those data files for file pruning

Apache Iceberg Benefits

Better Experience

Iceberg **abstracts** the physical representation of the data, making the lake look like any other database

- **Concurrent reads/writes** with serializable isolation
- **Hidden partitioning** so users don't have to know the partition columns
- **Rich schema evolution** support
- **Data is automatically optimized and rewritten** as needed
- **Change data capture** is much easier

Higher Performance

Iceberg has the **metadata** needed to accelerate query planning &execution (schemas, file pointers, column stats, etc.)

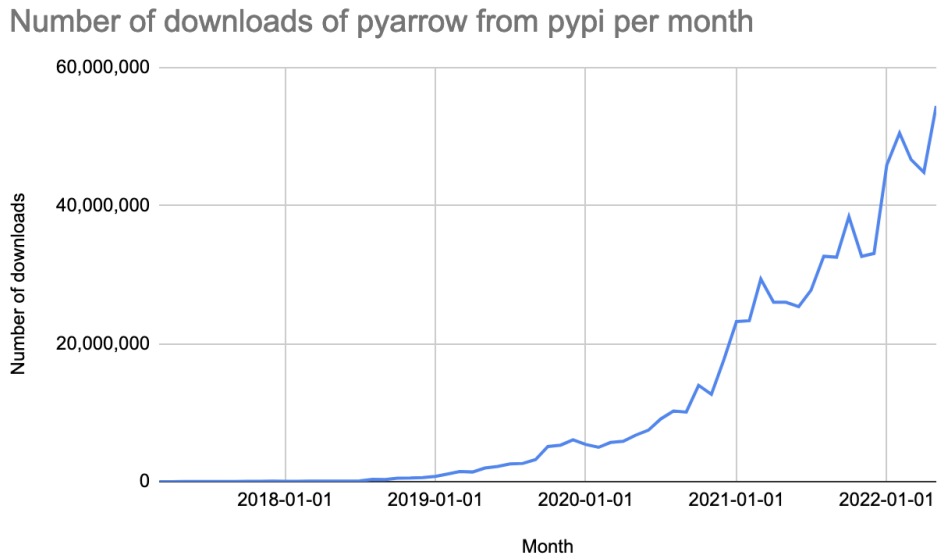
- **Efficient updates**
  - Copy-on-write (rewrite a data file)
  - Merge-on-read (write a delete file)
- **Enhanced cost-based optimization** via reliable statistics
- **Less data to read** by pruning data files based on statistics in manifest files
- **Lower latency** by eliminating S3/ADLS bucket lists

What is Apache Arrow

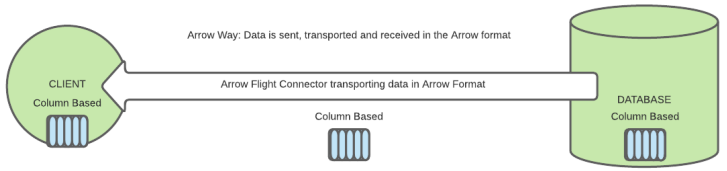
Open Table Format for Database Like Tables on The Lake

Introduction to Apache Arrow

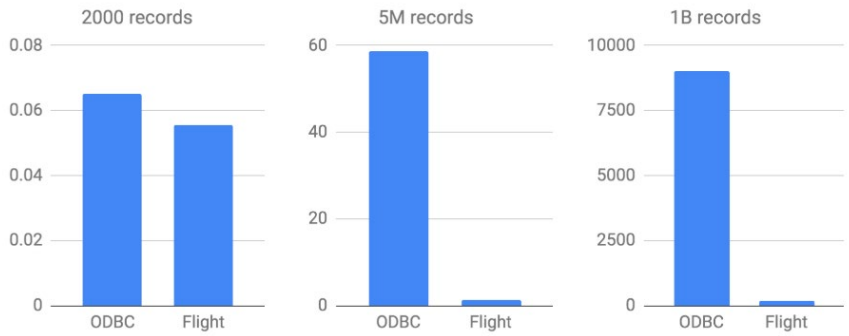
- An in-memory columnar format
  - Includes libraries for working with the format
    - E.g., computation engine, IPC, serialization / deserialization from file formats.
  - Support for many languages (12 currently)
- Co-created by Dremio and Wes McKinney
- Rapid adoption, popularity, and capability growth



What is Arrow Flight?



- A general-purpose client-server framework to **simplify high performance transport of large datasets over network interfaces**
- Designed for data in the modern world
  - **Column-oriented**, rather than row-oriented
  - Arrow's columnar format is designed for **high compressibility and large numbers of rows**
- Makes including the client-side in distributed computing easier
- Enables **parallel data transfers**



dremio

What is Arrow Flight SQL?

- **Client-server protocol for interacting with SQL databases built on Arrow Flight**
  - Allows databases to use Arrow Flight as the transport protocol
  - Leverage the performance of Arrow and Flight for database access

- **Set of commands to standardize a SQL interface on Flight:**
  - Query execution
  - Prepared statements
  - Database catalog metadata (tables, columns, data types).
  - SQL syntax capabilities

- **Language and database-agnostic**
  - A single set of client libraries to connect to any Flight SQL server
  - Clients can talk to any database implementing the protocol
  - No need for a client-side driver per database

Arrow Flight SQL ODBC & JDBC Drivers (+ ADBC)

- ODBC & JDBC Drivers built on top of Flight SQL client libraries
  - A single driver to connect to any Flight SQL server
  - Supports arbitrary server-side options as connection properties
- **The drivers will work with any ODBC/JDBC client tool without any code changes**
- Completely open source
- New ADBC Protocol

What is Nessie

Open Source Catalog for Cutting Edge Data Lakehouses

The Data Lake Evolution

Traditional Data Lake → Lakehouse → ???

Columnar File Formats (Apache Parquet)

- Multi-engine
- Scalable
- Fast queries

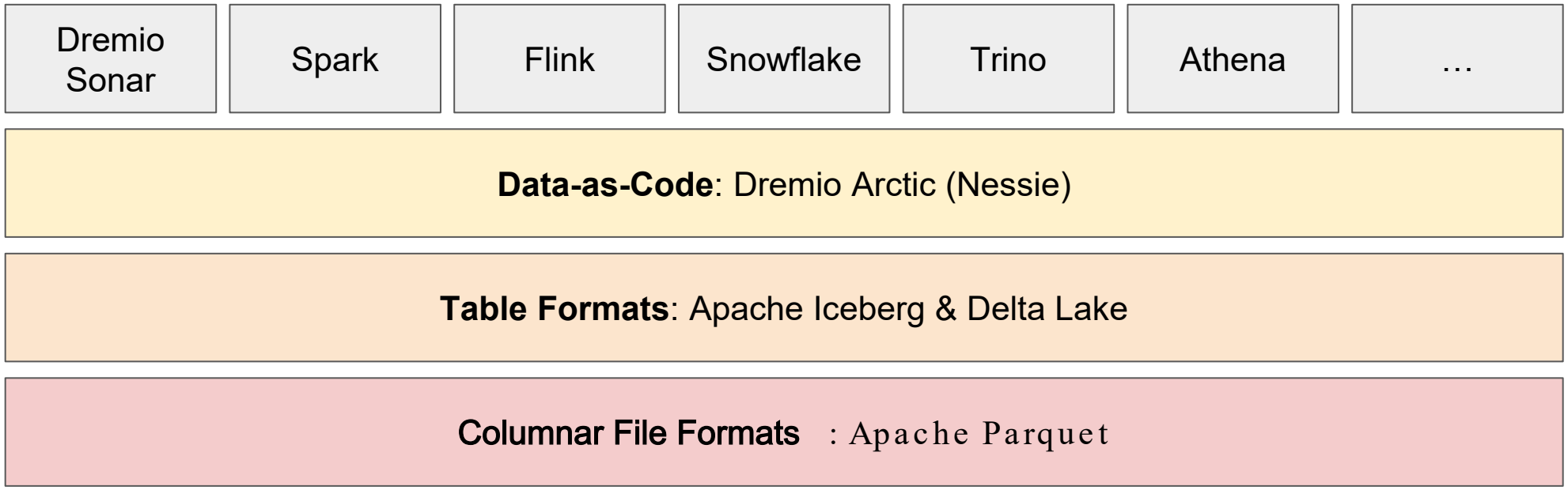
Open Table Formats (Apache Iceberg, Delta Lake)

- Read/write (DML)
- Schema/partition evolution
- Data optimization

Data-as-Code (Nessie)

- Isolation (branches)
- Version control
- Governance

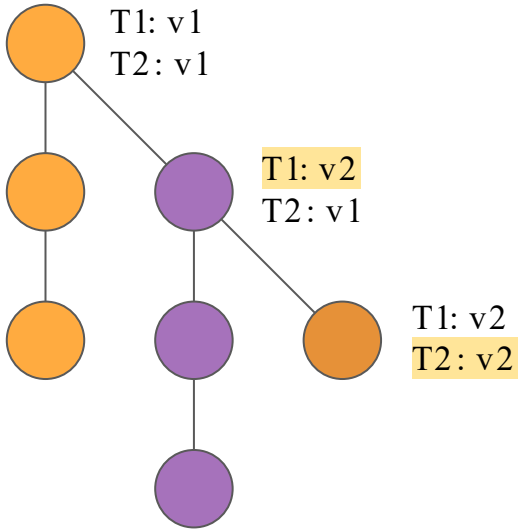
The Data Lake Evolution



The Data Lake Evolution

Engines: | Dremio Sonar | Spark | AWS (EMR, Athena, Spectrum) | ...

Nessie



Object storage (S3, Azure Storage, GCS)

Table: **T1**

Iceberg metadata files

v1 v2

Parquet data files

Table: **T2**

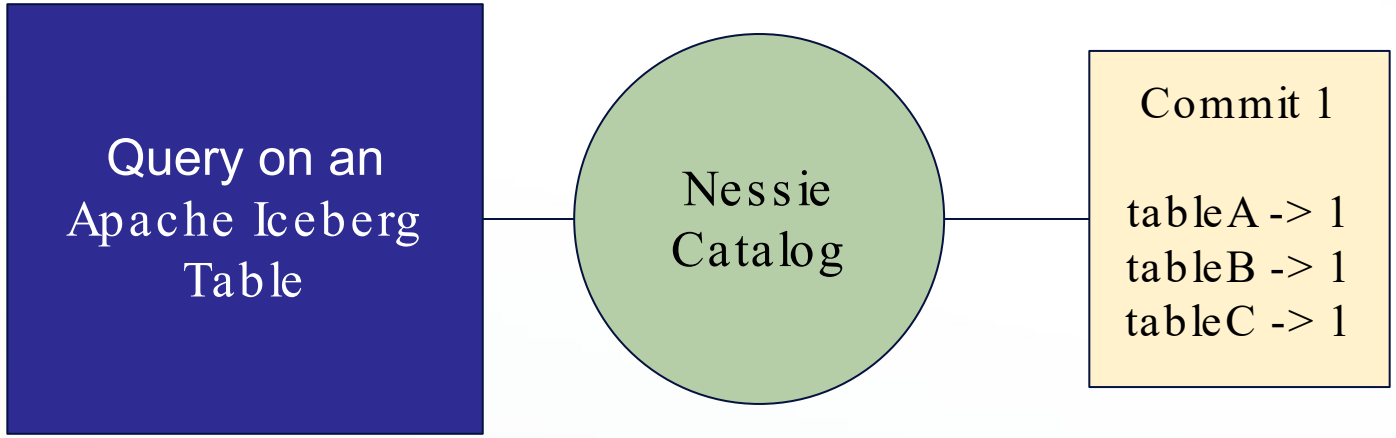
Iceberg metadata files

v1 v2

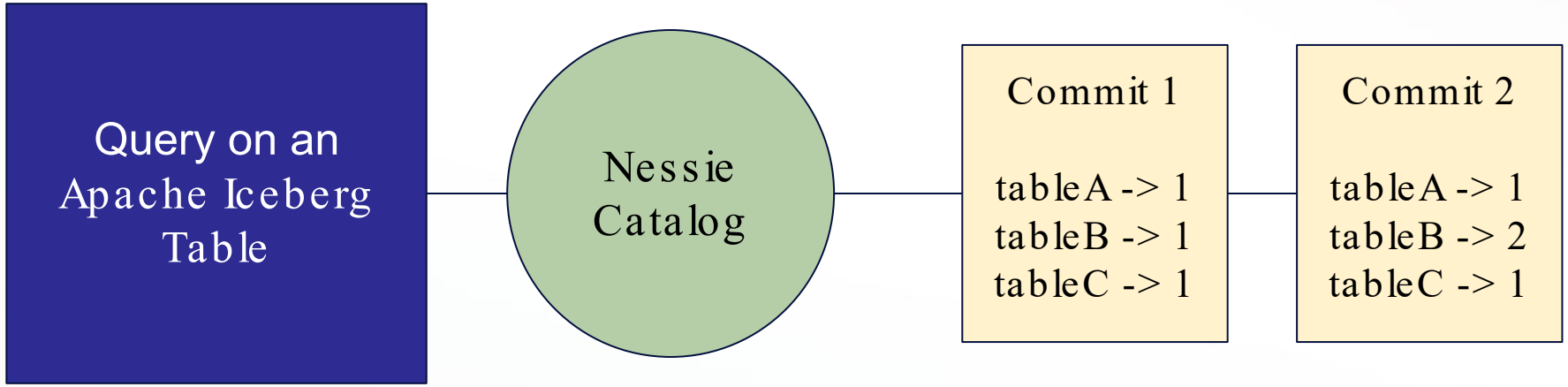
Parquet data files

dremio

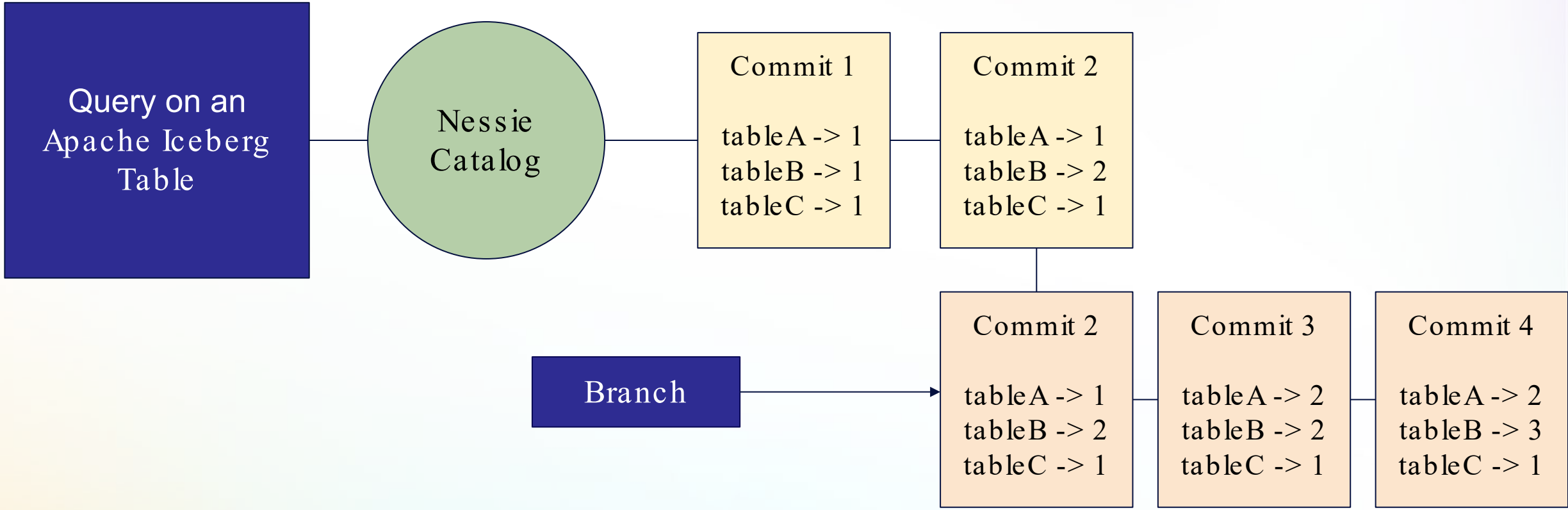
What is Project Nessie?



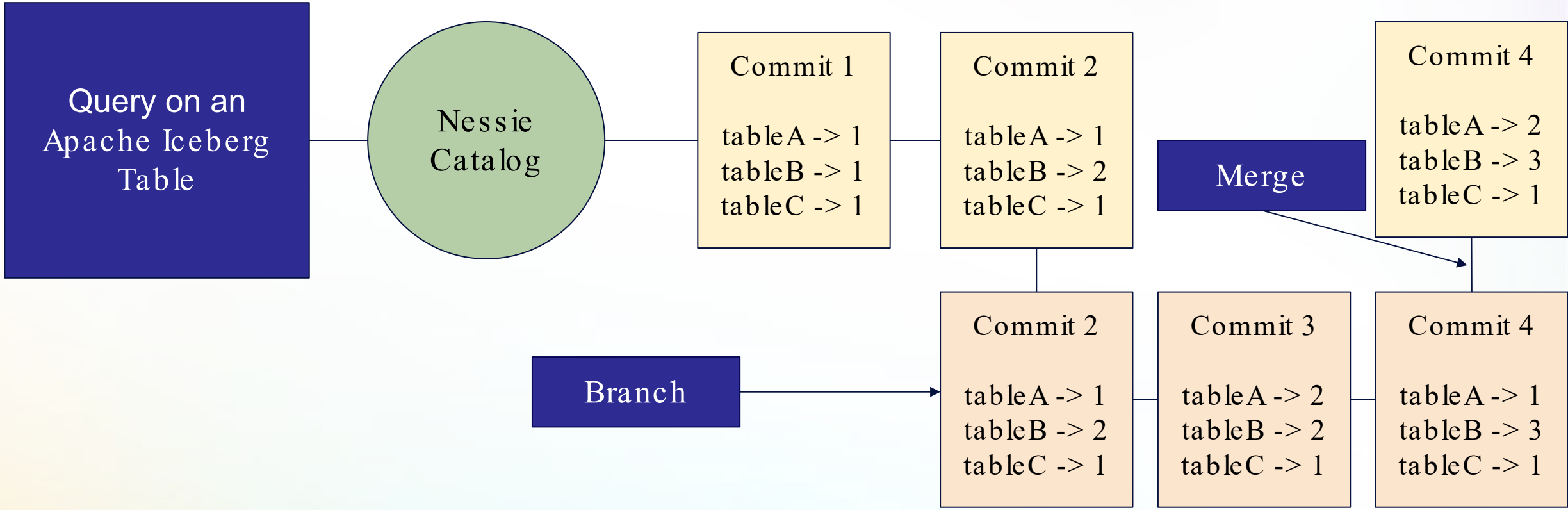
What is Project Nessie?



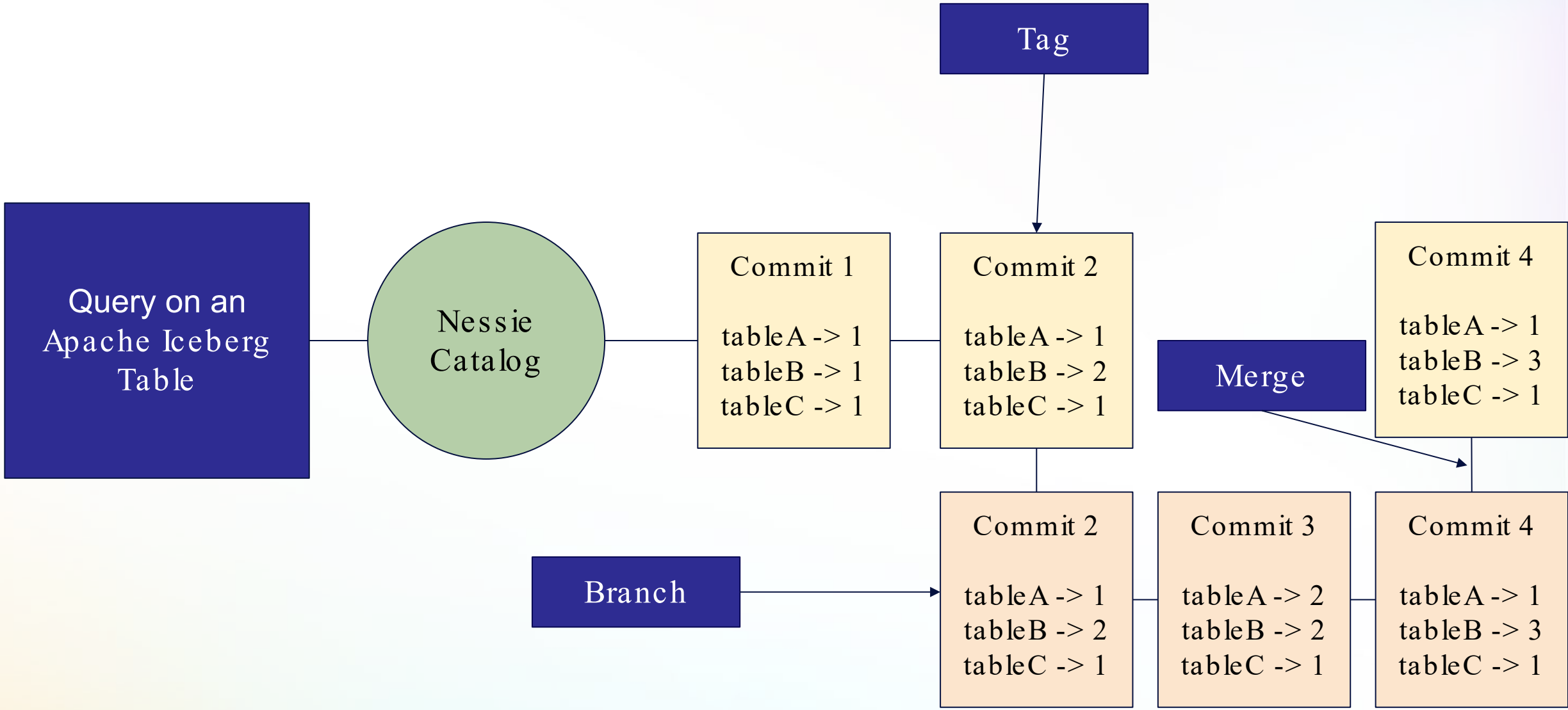
What is Project Nessie?



What is Project Nessie?



What is Project Nessie?



dremio

Benefits of Nessie

Catalog Level

Open Source

Managed Service

Cloud Agnostic

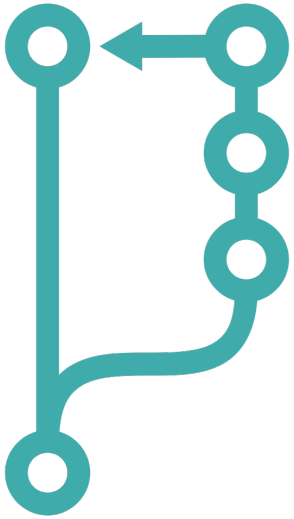
Storage Agnostic

Optimization

Nessie Introduces Data-as-Code

ISOLATION

- Experiment with data without impacting other users
- Ingest, transform and test data before exposing it to other users in an atomic merge



VERSION CONTROL

- Reproduce models and dashboards from historical data based on time or tags
- Recover from any mistake by instantly undoing accidental data or metadata changes

GOVERNANCE

- All changes to the data and metadata are tracked: who accessed what data and when
- Control table metadata access

Data Warehouse Transactions

One session
One user
SQL-only

Built for application developers

```
BEGIN TRANSACTION etl;
INSERT INTO t1 ...
UPDATE t1 ...
UPDATE t1 ...
COMMIT;
```

VS.

Real-World Transactions

Multiple sessions
Multiple users
Multiple engines/services

Built for data engineers

```
CREATE BRANCH etl
[Kafka]  Ingest data source 1
[RDBMS]  Ingest data source 2
[Spark]  Transformation step 1
[Dremio] Transformation step 2
[Spark]  Insert into table
[Dremio] Build reflections
[Spark]  Verify data
USE BRANCH main
MERGE BRANCH etl
```

Nessie Works with All Engines

Dremio Sonar

```
CREATE BRANCH etl_030222
USE BRANCH etl_030222
INSERT INTO t1 …
UPDATE t2 SET …
MERGE BRANCH etl_030222 INTO main
```

Apache Spark

```
spark.sql("CREATE BRANCH etl_030222 IN
arctic")
spark.sql("USE REFERENCE etl_030222 IN
arctic")
df = spark.read.parquet("...")
df.writeTo("arctic.t1").append()
spark.sql("UPDATE arctic.t2 SET …")
spark.sql("MERGE BRANCH etl_030222 INTO main
IN arctic")
```

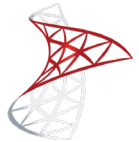
Apache Flink

```
arctic.create_branch("etl_030222", hash)

env.execute_sql("CREATE CATALOG …")

data = env.from_path("...")
data.execute_insert("arctic.t1").wait()

t2 = env.from_path("arctic.t2")
t2_mod = t2.filter(fil).map(mapf)
env.create_temporary_view("t2_mod", t2_mod)
env.execute_sql("MERGE INTO …")

arctic.merge(from_ref="etl_030222",
onto_branch="main")
```

Hands-On

Intro to Iceberg

Postgres -> Dashboard



SQLServer -> Dashboard

MongoDB -> Dashboard

dremio.com/blog

dremio

Be Part of the Data Lakehouse Community

The Data Lakehouse Hub

Join the Channel for your community

#meetup-orlando
#meetup-miami
#meetup-austin
#meetup-seattle
#meetup-denver
#meetup-miami
#meetup-nyc
#meetup-atlanta
#meetup-chicago
#meetup-san-francisco
#meetup-santa-clara

https://bit.ly/lakehouse-hub-slack

Never Miss an Event and Subscribe to the Calendar

https://lu.ma/Lakehouselinkups

Please take a moment to rate this session.

Your feedback is important to us.

