

# Apache Iceberg's REST Catalog

A Gateway to Enriching Data Access via the Simplicity of an HTTP Service



Samuel Redai - Developer Advocate, Tabular

**Samuel Redai - Developer Advocate, Tabular**

Twitter: [@samuelredai](#)

GitHub: [samredai](#)

Email: [sam@tabular.io](mailto:sam@tabular.io)



# What is Apache Iceberg?

**“Iceberg is a high-performance format for huge analytic tables. Iceberg brings the reliability and simplicity of SQL tables to big data, while making it possible for engines like Spark, Trino, Flink, Presto, and Hive to safely work with the same tables, at the same time.”**

- [iceberg.apache.org](https://iceberg.apache.org)

**Iceberg provides massive scale cloud-native SQL tables that are accessible by many compute engines.**

-me

Supported by many compute engines & platforms



trino



amazon  
ATHENA



amazon  
EMR



Flink



Starburst



snowflake



StarRocks



HIVE



Apache  
Impala

presto



dremio



databricks

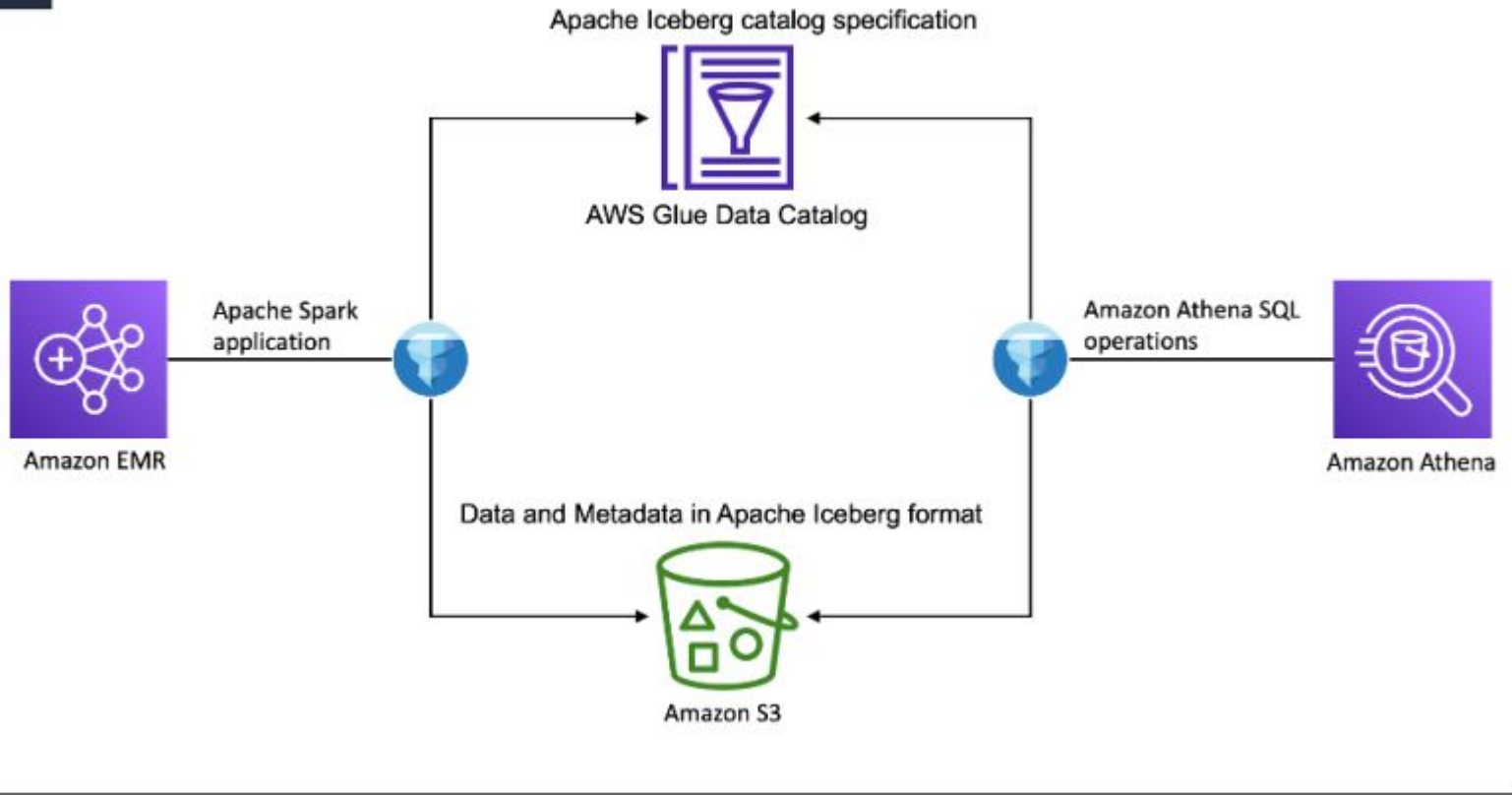
bodo.ai



Tabular



AWS Cloud



# What is a Catalog?



# Many ways to address a dataset...

Postgres:      <database> . <schema> . <table>

Hive:            <database> . <table>

Spark:          <catalog> . <database> . <table>

**Iceberg:**      **<namespace . . . > . <table>**

# Catalogs handle:

- Listing databases/tables
- Creating tables
- Dropping tables
- Renaming tables
- Providing atomicity to commits
- Transactions

# Integrating multiple compute engines

- Common interface
- Consistent behavior
- Manage contention

# Current Catalog Implementations

- HiveCatalog
- JdbcCatalog
- HadoopCatalog
- GlueCatalog
- NessieCatalog
- DynamoCatalog

# Challenges

- Consistency across implementations
- Client is responsible for compatibility
- New languages / ecosystems introduce more complexity

# Iceberg REST Catalog

# The REST Catalog

- A REST implementation of the Iceberg catalog specification
- The client makes REST requests to a server-side catalog
- The server is responsible for applying those commits and updating snapshot pointers
- Catalog specific dependencies are only required by the server
- The REST catalog server can wrap any existing Iceberg catalog implementation as well as include any additional server-side logic

# REST Catalog Features

- Standardized interface
- Centralized metadata management
- Multiple client versions
- Deconflict commit contention
- Improved security
  - authorization
  - identity aware sessions
- Connectivity



## Configuration API

**GET** /v1/config List all catalog configuration settings

## OAuth2 API

**POST** /v1/oauth/tokens Get a token using an OAuth2 flow

## Catalog API

**GET** /v1/{prefix}/namespaces List namespaces, optionally providing a parent namespace to list underneath

**POST** /v1/{prefix}/namespaces Create a namespace

**GET** /v1/{prefix}/namespaces/{namespace} Load the metadata properties for a namespace

**DELETE** /v1/{prefix}/namespaces/{namespace} Drop a namespace from the catalog. Namespace must be empty.

**POST** /v1/{prefix}/namespaces/{namespace}/properties Set or remove properties on a namespace

**GET** /v1/{prefix}/namespaces/{namespace}/tables List all table identifiers underneath a given namespace

**POST** /v1/{prefix}/namespaces/{namespace}/tables Create a table in the given namespace

**GET** /v1/{prefix}/namespaces/{namespace}/tables/{table} Load a table from the catalog

**POST** /v1/{prefix}/namespaces/{namespace}/tables/{table} Commit updates to a table

**DELETE** /v1/{prefix}/namespaces/{namespace}/tables/{table} Drop a table from the catalog

**HEAD** /v1/{prefix}/namespaces/{namespace}/tables/{table} Check if a table exists

**POST** /v1/{prefix}/tables/rename Rename a table from its current name to a new name

**POST** /v1/{prefix}/namespaces/{namespace}/tables/{table}/metrics Send a metrics report to this endpoint to be processed by the backend



# Non-REST Catalog

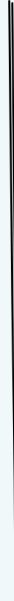
Append



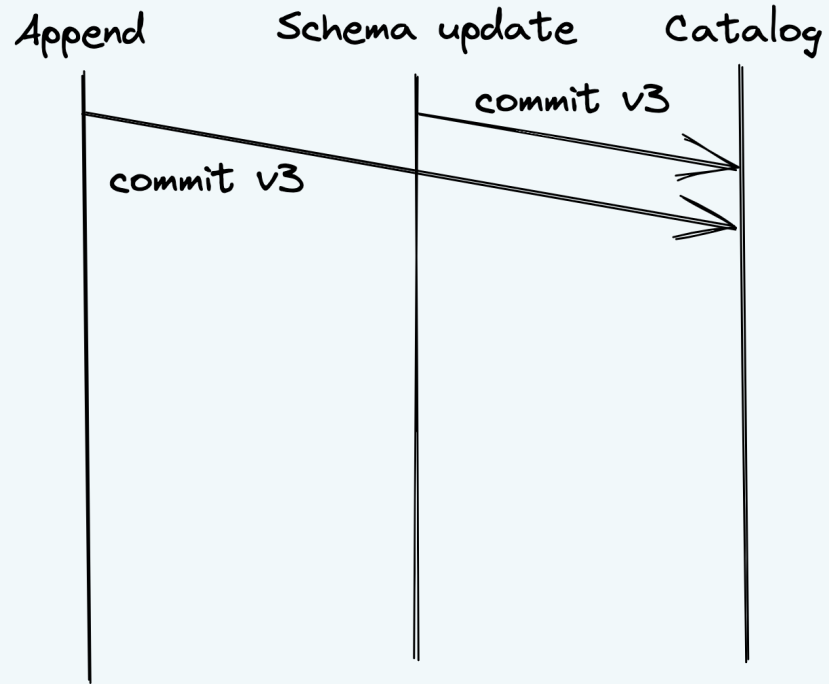
Schema update



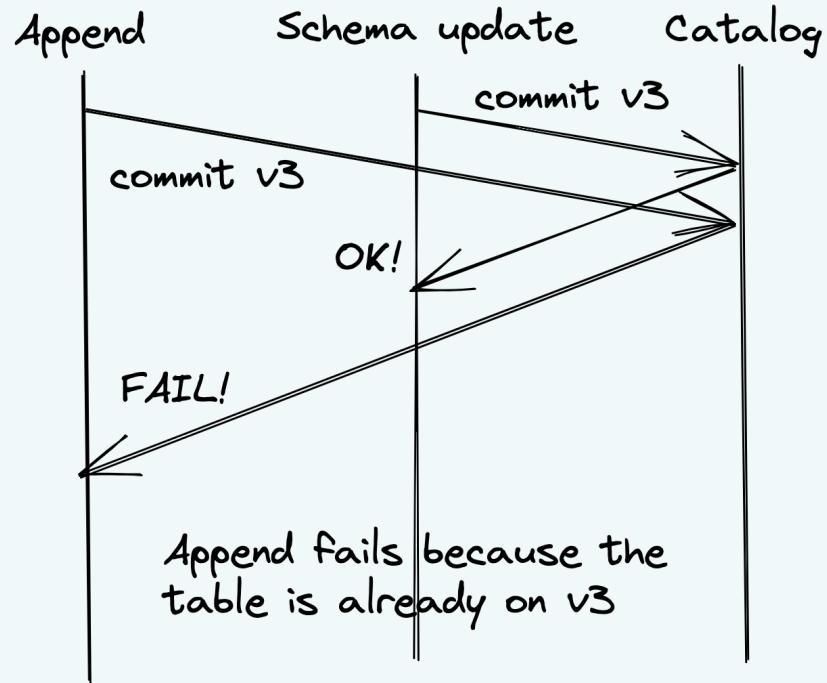
Catalog



# Non-REST Catalog



# Non-REST Catalog



# REST Catalog

Append



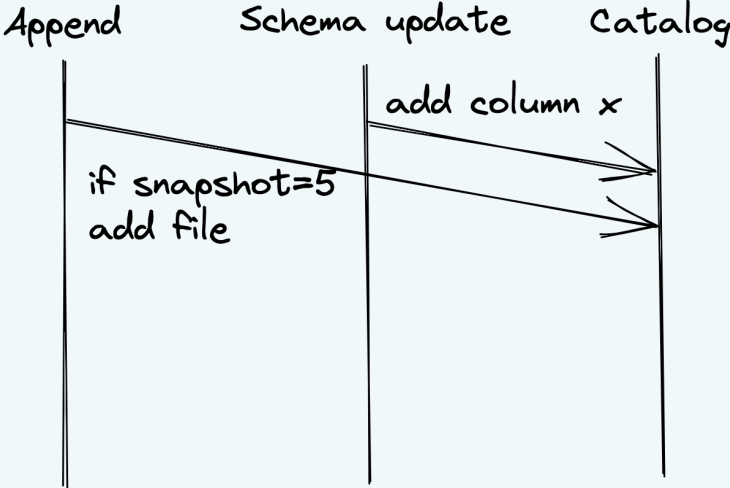
Schema update



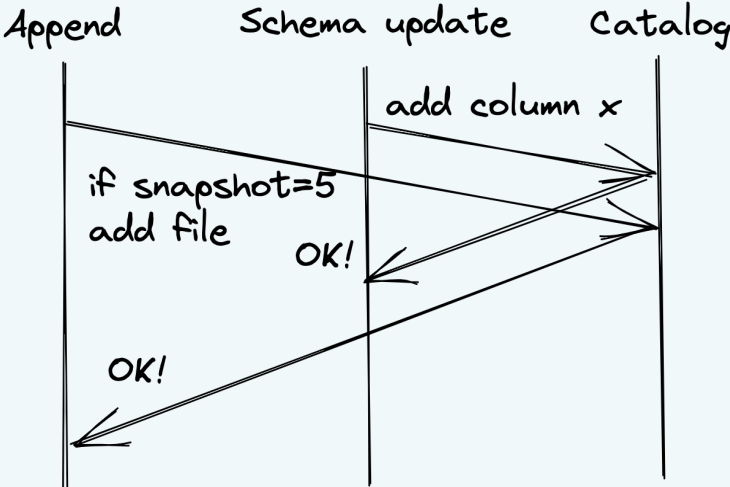
Catalog



# REST Catalog

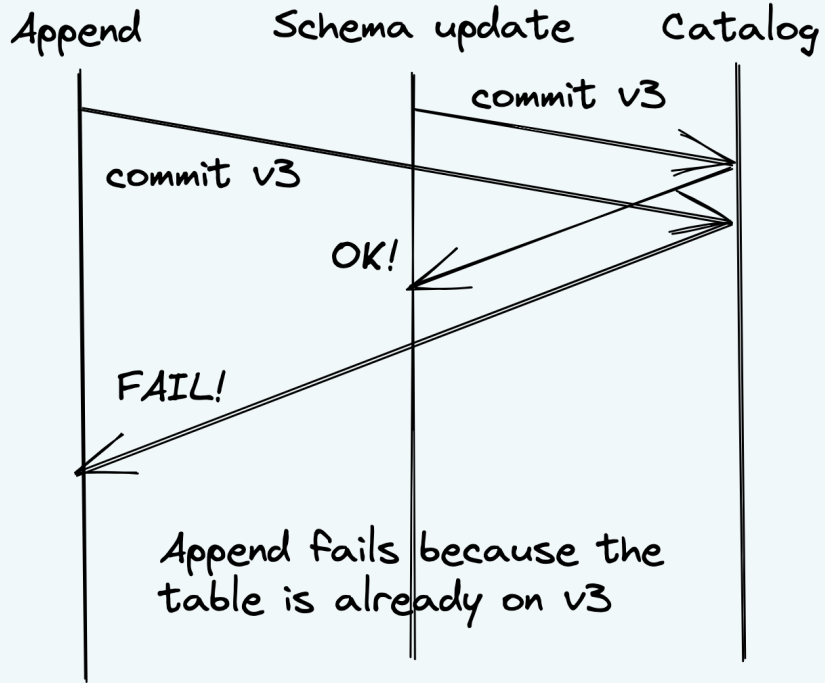


# REST Catalog

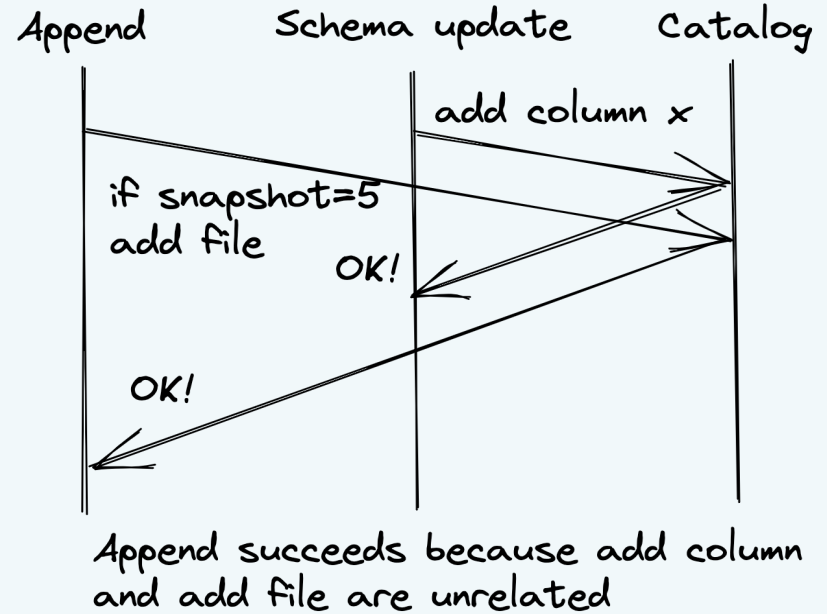


Append succeeds because add column and add file are unrelated

## Non-REST Catalog



## REST Catalog





# Pyiceberg Demo

by Fokko Driesprong

# REST Catalog – What's Next?

- Multi-table transactions
- Scan and commit metrics
- View catalog

# Thank You

- Check us out at [tabular.io](https://tabular.io)
- Join the Iceberg Community
  - [iceberg.apache.org/community](https://iceberg.apache.org/community)

