

Amazon Athena ACID transaction workshop

2025/01/31

シニアエバンジェリスト 亀田

はじめに：

Amazon Athena はサーバレスで S3 などのデータに対して SQL クエリーが発行可能なインタラクティブな分析サービスです。従来、Athena は SELECT 及び CTAS (Create Table As Select) のみに対応していましたが、新しく ACID トランザクションをサポートし、INSERT, DELETE, UPDATE、よして過去のデータに対してクエリーが発行可能なタイムトラベルクエリに対応しました。この機能を使うためには、S3 に保存されているデータは Apache Iceberg 形式である必要があります。

1. S3 のマネージメントコンソールに移動します
2. [バケットの作成]をおします



3. 適当な名前を入力しその他はデフォルトのまま[バケットを作成]をおします。作成したバケットの名前はコピーしておいてください
4. Athena のマネージメントコンソールを開きワークグループをクリックします。ワークグループとはユーザーごとの作業環境でありクエリの発行結果や一時計算に用いられる S3 バケットを指定します。



5. [ワークグループを作成]をおします
6. 適当な名前を設定します

ワークグループの詳細

ワークグループの一意の名前を入力します。ワークグループ名を変更するには、ワークグループを削除し、新しい名前で作成します。

ワークグループ名

iceburg20220415

ワークグループ名は一意である必要があります。1~128 文字を使用します。(A~Z、a~z、0~9、_、-、.)。作成後に名前を変更することはできません。

説明 - オプション

最大 1024 文字です。1024 文字残り。

7. 先程作成したバケットを指定し [/result/]を付与します

クエリ結果の設定

クエリ結果の場所

クエリ結果がオブジェクトとして保存される現在のリージョンの S3 プレフィックスを入力します。

🔍

s3://20220415hkamedaathena/result/

✕

表示 🗨

S3 を参照

8. [AWS CloudWatch にクエリメトリクスを発行]のチェックを外し、[上書き]のチェックをつけます

設定

メトリクス

クエリの成功、ランタイム、およびメトリクスタブでスキャンされたデータ量を表示します。Athenaはこれらのメトリクスを CloudWatch に送信します。

☐ AWS CloudWatch にクエリメトリクスを発行

上書き | 情報

これがチェックされていると、ワークグループ設定がワークグループ内のすべてのクエリに適用されます。

☒ クライアント側の設定を上書き

リクエスト | 情報

有効にすると、ワークグループユーザーはリクエスト支払いバケットでクエリできます。データリクエストと転送に対する支払いはユーザーのアカウントから行います。無効にすると、ワークグループユーザーのリクエスト支払いバケットでのクエリは失敗します。

☐ Amazon S3 のリクエスト支払いバケットでクエリを有効化

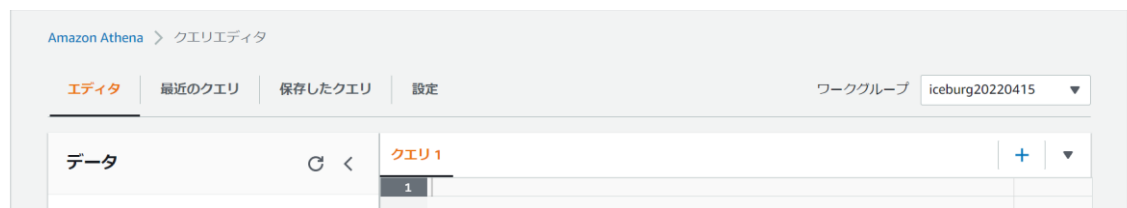
9. [ワークグループを作成]をおします。以下の通りワークグループが作成されていれば完了です



これで実行の準備ができました。早速クエリーを発行していきます。

1 0. 左ペインの[クエリエディタ]をおします

1 1. 画面右上のドロップダウンから先程作成した[ワークグループ]を選びます。追加で新しいウィンドウが表示された場合は、オレンジ色のボタンを押します



1 2. 以下のクエリーを 2 回に分けて実行します。<mybucket>は先程作成した S3 バケットへ置き換えてください

1 回目：

```
CREATE DATABASE IF NOT EXISTS my_database;
```

2 回目：

```
CREATE TABLE iceberg_table20220415 (
  id int,
  data string,
  category string)
PARTITIONED BY (category, bucket(16,id))
LOCATION 's3://<mybucket>/iceberg_table/'
TBLPROPERTIES (
  'table_type'='ICEBERG',
```

```
'format'='parquet',
'write_target_data_file_size_bytes'='536870912'
)
```

```
1 CREATE TABLE iceberg_table (
2   id int,
3   data string,
4   category string)
5 PARTITIONED BY (category, bucket(16,id))
6 LOCATION 's3://20220415hkamedaathena/iceberg_table/'
7 TBLPROPERTIES (
8   'table_type'='ICEBERG',
9   'format'='parquet',
10  'write_target_data_file_size_bytes'='536870912'
11 )
```

SQL 行 1、列 1

もう一度実行する キャンセル 保存 ▼ クリア 作成 ▼

- 1 3. テーブルができましたので以下のクエリーを順番に 1 行ずつじっこうします

```
INSERT INTO my_database.iceberg_table20220415 (id, data, category) values(1,
'Amazon', 'Athena');
INSERT INTO my_database.iceberg_table20220415 (id, data, category) values(2,
'Amazon', 'Redshift');
```

- 1 4. 以下のクエリーで、テーブルの中身を確認します。2 行 Insert が成功していることがわかります。

```
SELECT * from my_database.iceberg_table20220415;
```

完了	キュー内の時間: 0.199 秒	実行時間: 4.387 秒	スキャンしたデータ: 0.21 KB
結果 (2)			
<div> <div>コピー</div> <div>結果をダウンロード</div> </div> <div> <div>行を検索</div> <div>< 1 ></div> </div>			
#	id	data	category
1	2	Amazon	Redshift
2	1	Amazon	Athena

- 1 5. 次にタイムトラベルクエリを試してみます。以下のクエリーは上で実行したクエリーを 1 日前のデータに対して実行します

```
SELECT * FROM my_database.iceberg_table20220415 FOR VERSION AS OF
```

(current_timestamp - interval '1' day);

1 日前はテーブルが存在していないため、以下の通りエラーが発生します

❌ 失敗

キュー内の時間: 0.263 秒

実行時間: 2.505 秒

スキャンしたデータ: -

❌ GENERIC_USER_ERROR: No table snapshot found before timestamp 1649914746842

このクエリは、クエリで修飾されていない限り、「default」データベースに対して実行されました。エラーメッセージを [フォーラム](#) に投稿するか、クエリ ID: 281e2cce-89d9-418b-b8e5-cacdac5ad0e5 とともに [カスタマーサポート](#) にお問い合わせください。

タイムトラベルクエリの詳細はこちらをご覧ください

<https://docs.aws.amazon.com/athena/latest/ug/querying-iceberg-table-data.html>

1 6. 今度は Update クエリーを実行してみます

```
UPDATE my_database.iceberg_table20220415 SET data='Redshift Spectrum' WHERE  
id = 2;
```

1 7. 全件検索をかけると以下の通り値が書き換わっていることがわかります

```
SELECT * from my_database.iceberg_table20220415;
```

もう一度実行する					キャンセル	保存 ▼	クリア	作成 ▼
✅ 完了					キュー内の時間: 0.127 秒	実行時間: 5.34 秒	スキャンしたデータ: 0.32 KB	
結果 (2)					📄 コピー		結果をダウンロード	
🔍 行を検索					< 1 >		⚙️	
# ▼	id ▼	data ▼			category ▼			
1	2	Redshift Spectrum			Redshift			
2	1	Amazon			Athena			

1 8. 次に以下のクエリーでレコードの削除を行います

```
DELETE FROM my_database.iceberg_table20220415 WHERE id=1;
```

1 9. 全件検索をかけると以下の通りレコードが 1 件削除されていることがわかります

```
SELECT * from my_database.iceberg_table20220415;
```

<div> <div>もう一度実行する</div> <div>キャンセル</div> <div>保存 ▼</div> <div>クリア</div> <div>作成 ▼</div> </div>			
<div> <div>完了</div> <div>キュー内の時間: 0.136 秒</div> <div>実行時間: 5.235 秒</div> <div>スキャンしたデータ: 0.32 KB</div> </div>			
<div> <div>結果 (1)</div> <div> <div>コピー</div> <div>結果をダウンロード</div> </div> </div>		<div> <div>行を検索</div> <div> <div><</div> <div>1</div> <div>></div> <div>設定</div> </div> </div>	
# ▼	id ▼	data ▼	category ▼
1	2	Redshift Spectrum	Redshift

では最後に以下のクエリを数分ずつ遡って実行してください。通常では出てこない過去のスナップショットに対するクエリ結果が出てきます。

```
SELECT * FROM iceberg_table20220415 FOR TIMESTAMP AS OF TIMESTAMP
'2025-01-26 08:40:00 UTC'
```

おつかれさまでした！
削除は以下を行ってください
S3 バケット
Athena ワークグループ