rp6578

Data to Dashboards

Data to Dashboard - Real-time Data Processing and Analysis
Prerequisites
Introduction
Ingesting Real-time Data Streams
▼ Data Processing using Amazon Managed Apache Flink
Overview
Preparation
Run Studio Notebook
▼ Deliver Processed Data using Amazon Data Firehose
Overview
Deliver data
Access and Validate processed data
▶ Visualize Real-time data using Amazon QuickSight
Conclusion & Next Steps

▼ **AWS account access**

Open AWS console (us-east-1)
Get AWS CLI credentials

Exit event

Event dashboard > Deliver Processed Data using Amazon Data Firehose > Access and Validate processed data

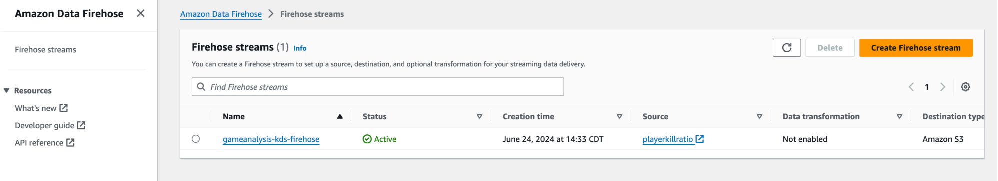
Access and Validate processed data

Now, Lets test the flow and validate the data.

In the **Data Processing using Amazon Managed Apache Flink** module we initiated ingestion of processed data into **playerkillratio** data stream.

Amazon Data Firehose will deliver this processed data to Amazon S3 as destination under the prefix provided. The buffer is set to 30 seconds or 5 MB, whichever happens first. It may be 30 seconds before you see data in S3.

1. Navigate to the Firehose Console and click on the **gameanalysis-kds-firehose** Data Firehose delivery stream.

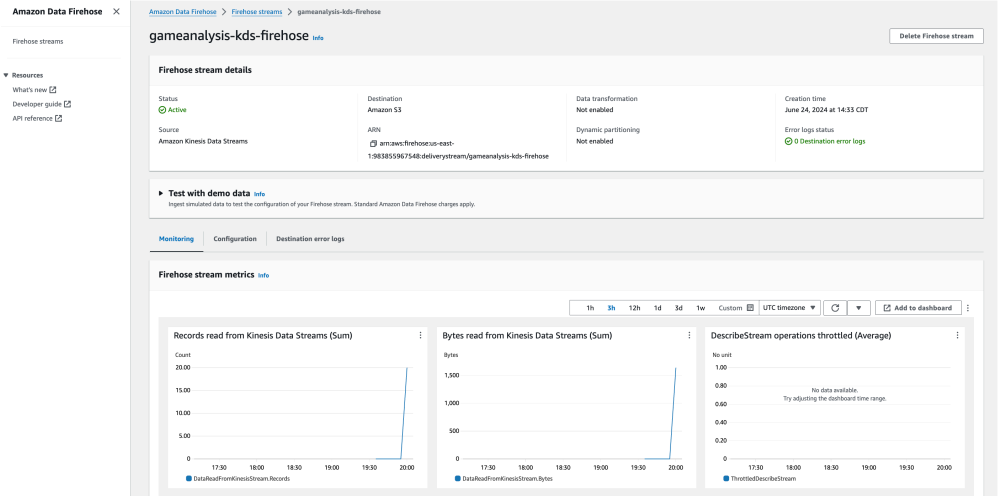


Firehose streams (1) Info

You can create a Firehose stream to set up a source, destination, and optional transformation for your streaming data delivery.

	Name	Status	Creation time	Source	Data transformation	Destination type
○	gameanalysis-kds-firehose	Active	June 24, 2024 at 14:33 CDT	playerkillratio	Not enabled	Amazon S3

2. Click on the Monitoring tab and you should start to see some metrics from your Firehose stream.



gameanalysis-kds-firehose Info

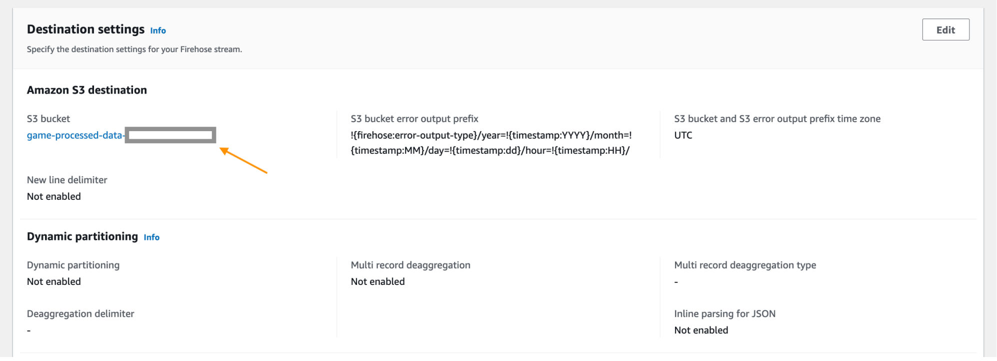
Firehose stream details

Status: Active
Source: Amazon Kinesis Data Streams
Destination: Amazon S3
ARN: arn:aws:firehose:us-east-1:983855967548:deliverystream/gameanalysis-kds-firehose
Data transformation: Not enabled
Dynamic partitioning: Not enabled
Creation time: June 24, 2024 at 14:33 CDT
Error logs status: 0 Destination error logs

▶ **Test with demo data** Info
Ingest simulated data to test the configuration of your Firehose stream. Standard Amazon Data Firehose charges apply.

Firehose stream metrics Info

3. Click on the Configuration tab. Scroll down to the Destination setting section. Then click on the S3 bucket link.



Destination settings Info

Specify the destination settings for your Firehose stream.

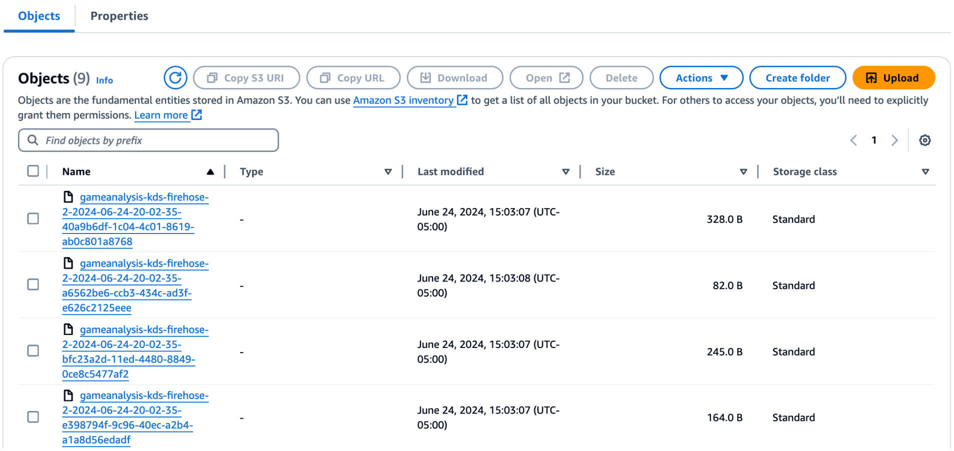
Amazon S3 destination

S3 bucket: game-processed-data-[redacted]
S3 bucket error output prefix: !{firehose:error-output-type}/year=!{timestamp:YYYY}/month=!{timestamp:MM}/day=!{timestamp:dd}/hour=!{timestamp:HH}/
S3 bucket and S3 error output prefix time zone: UTC
New line delimiter: Not enabled

Dynamic partitioning Info

Dynamic partitioning: Not enabled
Multi record deaggregation: Not enabled
Multi record deaggregation type: -
Deaggregation delimiter: -
Inline parsing for JSON: Not enabled

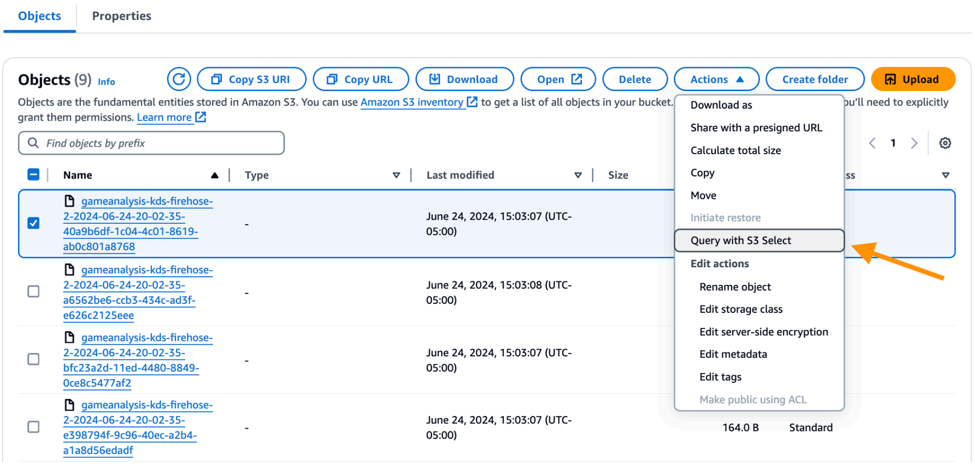
4. Give it a couple of minutes and you should see some data in your S3 bucket. If not, please wait a bit longer. Navigate into the game-processed-data folder and through the subfolders until you get to the data files in JSON format.



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Last modified	Size	Storage class
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-40a9b6df-1c04-4c01-8619-ab0c801a8768	-	June 24, 2024, 15:03:07 (UTC-05:00)	328.0 B	Standard
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-a6562be6-ccb3-434c-ad3f-e626c2125eee	-	June 24, 2024, 15:03:08 (UTC-05:00)	82.0 B	Standard
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-bfc23a2d-11ed-4480-8849-0ce8c5477af2	-	June 24, 2024, 15:03:07 (UTC-05:00)	245.0 B	Standard
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-e398794f-9c96-40ec-a2b4-a1a8d56edadf	-	June 24, 2024, 15:03:07 (UTC-05:00)	164.0 B	Standard

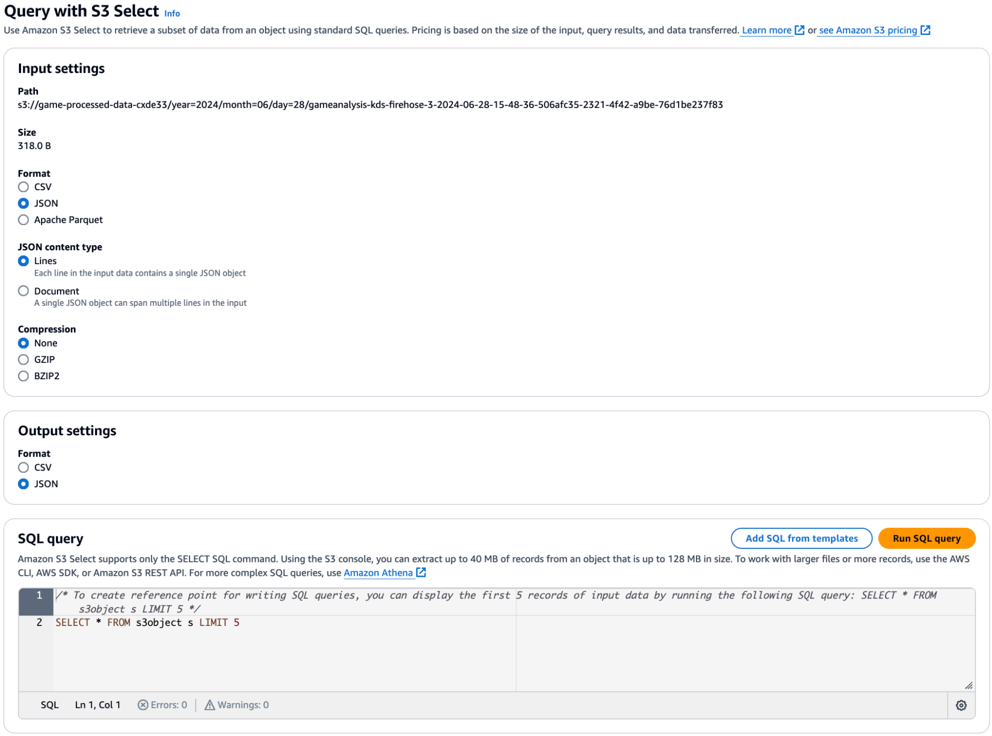
5. Select one of the file. Then under Actions, choose the Select Object Actions then Query with S3 Select.



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. ou'll need to explicitly grant them permissions. Learn more

☐	Name ▲	Type	Last modified	Size	ss
☑	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-40a9b6df-1c04-4c01-8619-ab0c801a8768	-	June 24, 2024, 15:03:07 (UTC-05:00)		
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-a6562be6-ccb3-434c-ad3f-e626c2125eee	-	June 24, 2024, 15:03:08 (UTC-05:00)		
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-bfc23a2d-11ed-4480-8849-0ce8c5477af2	-	June 24, 2024, 15:03:07 (UTC-05:00)		
□	gameanalysis-kds-firehose-2-2024-06-24-20-02-35-e398794f-9c96-40ec-a2b4-a1a8d56edadf	-	June 24, 2024, 15:03:07 (UTC-05:00)	164.0 B	Standard

6. Keep JSON as selected for Input settings. Choose JSON for Output Settings, then select Run SQL Query.



Query with S3 Select Info

Use Amazon S3 Select to retrieve a subset of data from an object using standard SQL queries. Pricing is based on the size of the input, query results, and data transferred. Learn more or see Amazon S3 pricing

Input settings

Path
s3://game-processed-data-cxde33/year=2024/month=06/day=28/gameanalysis-kds-firehose-3-2024-06-28-15-48-36-506afc35-2321-4f42-a9be-76d1be237f83

Size
318.0 B

Format
○ CSV
● JSON
○ Apache Parquet

JSON content type
● Lines
Each line in the input data contains a single JSON object
○ Document
A single JSON object can span multiple lines in the input

Compression
● None
○ GZIP
○ BZIP2

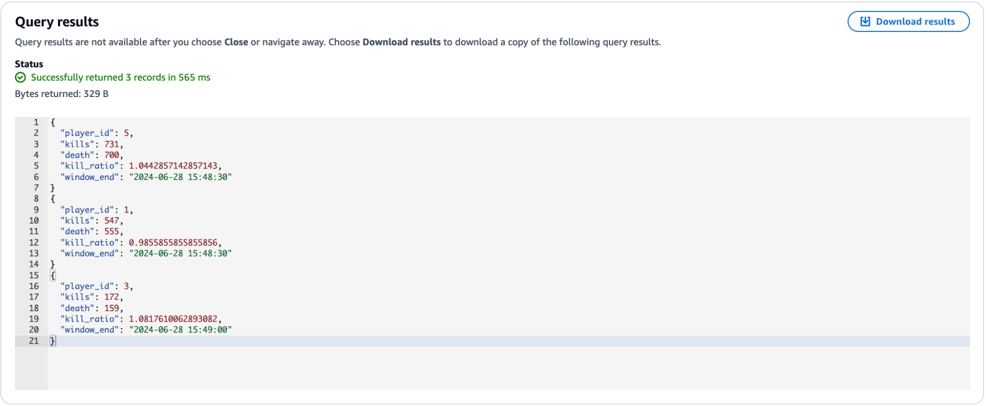
Output settings

Format
○ CSV
● JSON

SQL query

Amazon S3 Select supports only the SELECT SQL command. Using the S3 console, you can extract up to 40 MB of records from an object that is up to 128 MB in size. To work with larger files or more records, use the AWS CLI, AWS SDK, or Amazon S3 REST API. For more complex SQL queries, use Amazon Athena

```sql
/* To create reference point for writing SQL queries, you can display the first 5 records of input data by running the following SQL query: SELECT * FROM
    s3object s LIMIT 5 */
SELECT * FROM s3object s LIMIT 5
```



Query results

Query results are not available after you choose **Close** or navigate away. Choose **Download results** to download a copy of the following query results.

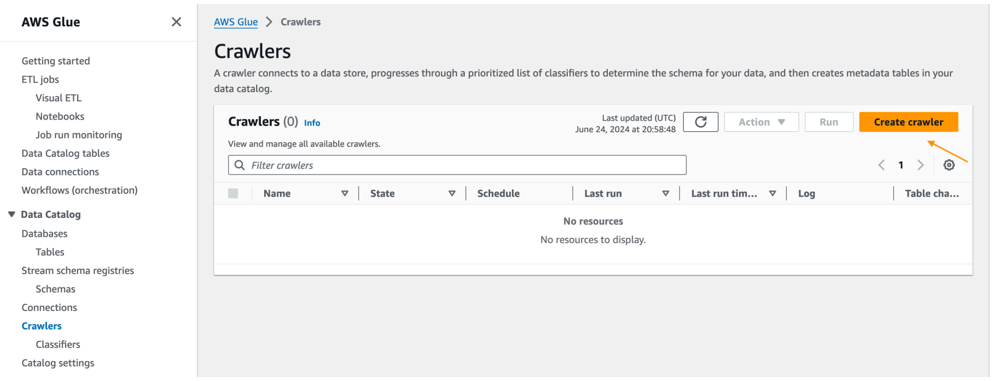
Status
Successfully returned 3 records in 565 ms
Bytes returned: 329 B

```json
{
  "player_id": 5,
  "kills": 731,
  "death": 700,
  "kill_ratio": 1.0442857142857143,
  "window_end": "2024-06-28 15:48:30"
}
{
  "player_id": 1,
  "kills": 547,
  "death": 555,
  "kill_ratio": 0.9855855855855856,
  "window_end": "2024-06-28 15:48:30"
}
{
  "player_id": 3,
  "kills": 172,
  "death": 159,
  "kill_ratio": 1.0817610062893082,
  "window_end": "2024-06-28 15:49:00"
}
```

The processed game data can be observed in the S3 bucket, indicating that the Amazon Data Firehose service has successfully completed its task.

AWS Glue Setup

1. Navigate to the AWS Glue Console. Choose Crawlers in the navigation pane.
2. Select **Create Crawler**.



Crawlers

A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.

Crawlers (0) Info

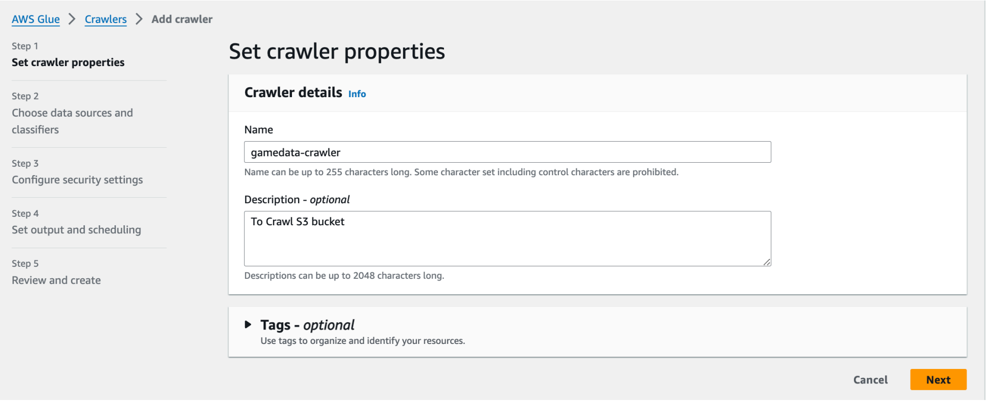
Last updated (UTC)
June 24, 2024 at 20:58:48

View and manage all available crawlers.

Name	State	Schedule	Last run	Last run tim...	Log	Table cha...

No resources
No resources to display.

3. Provide name of the crawler as **gamedata-crawler** and select **Next** .



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Set crawler properties

Crawler details Info

Name: gamedata-crawler

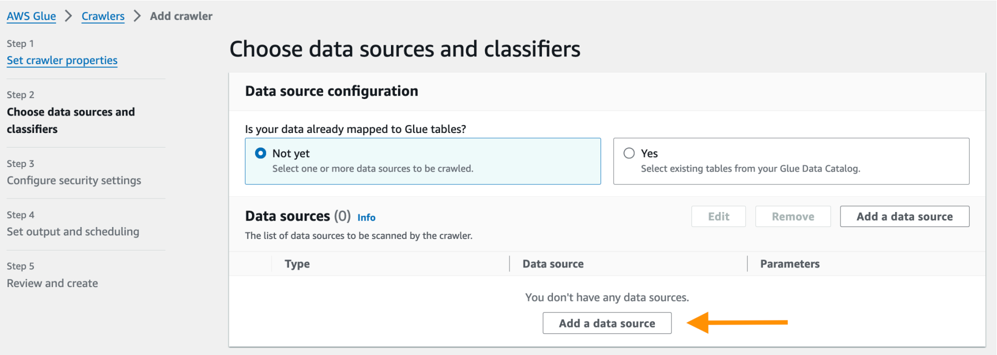
Name can be up to 255 characters long. Some character set including control characters are prohibited.

Description - *optional*: To Crawl S3 bucket

Descriptions can be up to 2048 characters long.

▶ **Tags - *optional***
Use tags to organize and identify your resources.

4. Click on the add a data source and browse the s3 bucket 'gameanalysis-kds-firehose-*'. Select **Next**.



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

● Not yet
Select one or more data sources to be crawled.

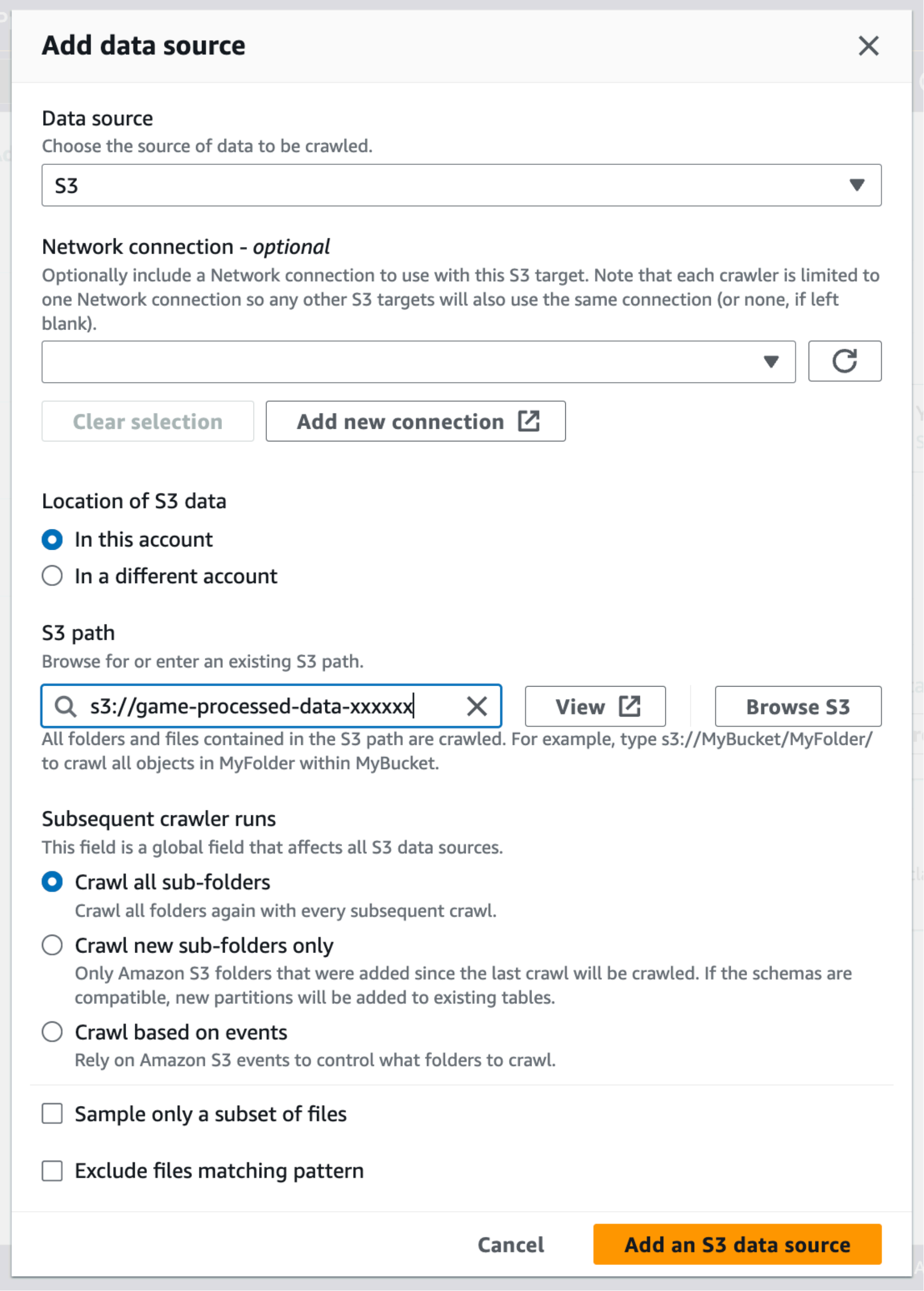
○ Yes
Select existing tables from your Glue Data Catalog.

Data sources (0) Info

The list of data sources to be scanned by the crawler.

Type	Data source	Parameters

You don't have any data sources.



Add data source

Data source
Choose the source of data to be crawled.
S3

Network connection - *optional*
Optionally include a Network connection to use with this S3 target. Note that each crawler is limited to one Network connection so any other S3 targets will also use the same connection (or none, if left blank).

Location of S3 data
● In this account
○ In a different account

S3 path
Browse for or enter an existing S3 path.
s3://game-processed-data-xxxxxx

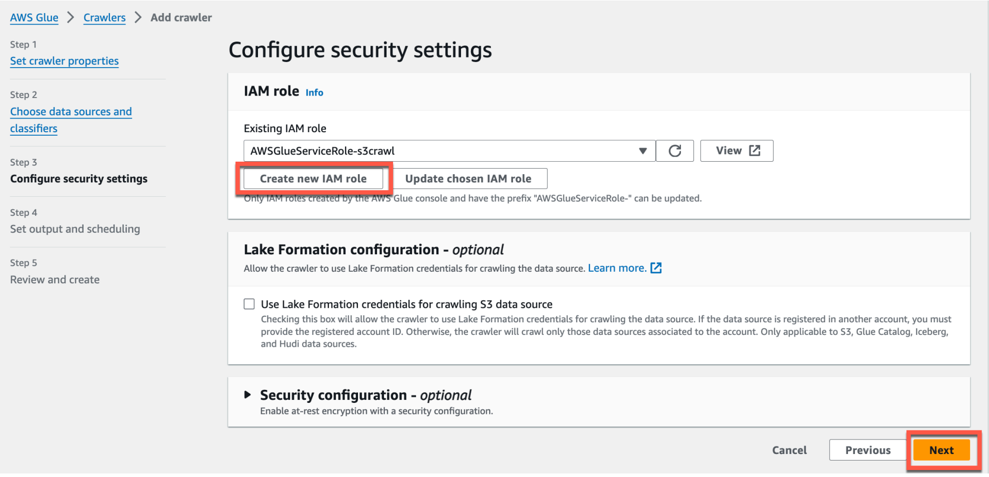
All folders and files contained in the S3 path are crawled. For example, type s3://MyBucket/MyFolder/ to crawl all objects in MyFolder within MyBucket.

Subsequent crawler runs
This field is a global field that affects all S3 data sources.
● Crawl all sub-folders
Crawl all folders again with every subsequent crawl.
○ Crawl new sub-folders only
Only Amazon S3 folders that were added since the last crawl will be crawled. If the schemas are compatible, new partitions will be added to existing tables.
○ Crawl based on events
Rely on Amazon S3 events to control what folders to crawl.

□ Sample only a subset of files

□ Exclude files matching pattern

5. Create a new IAM role and Select **Next**.



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Configure security settings

IAM role Info

Existing IAM role

AWSGlueServiceRole-s3crawl ▼

Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.

Lake Formation configuration - *optional*

Allow the crawler to use Lake Formation credentials for crawling the data source. Learn more.

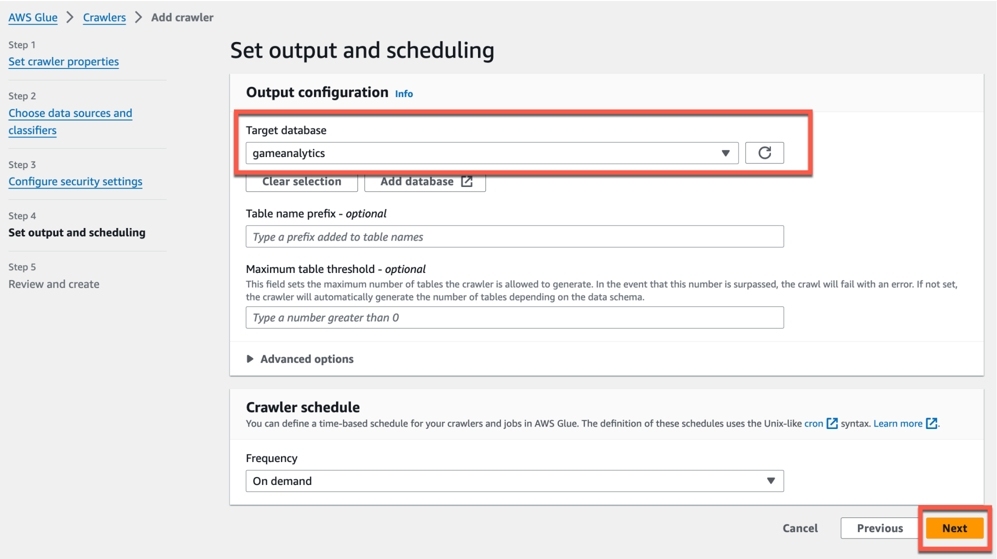
□ Use Lake Formation credentials for crawling S3 data source

Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source is registered in another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3, Glue Catalog, Iceberg, and Hudi data sources.

▸ **Security configuration** - *optional*

Enable at-rest encryption with a security configuration.

6. Choose **gameanalytics** as as database and Select **Next**.



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Set output and scheduling

Output configuration Info

Target database

gameanalytics ▼

Table name prefix - *optional*

Type a prefix added to table names

Maximum table threshold - *optional*

This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.

Type a number greater than 0

▸ **Advanced options**

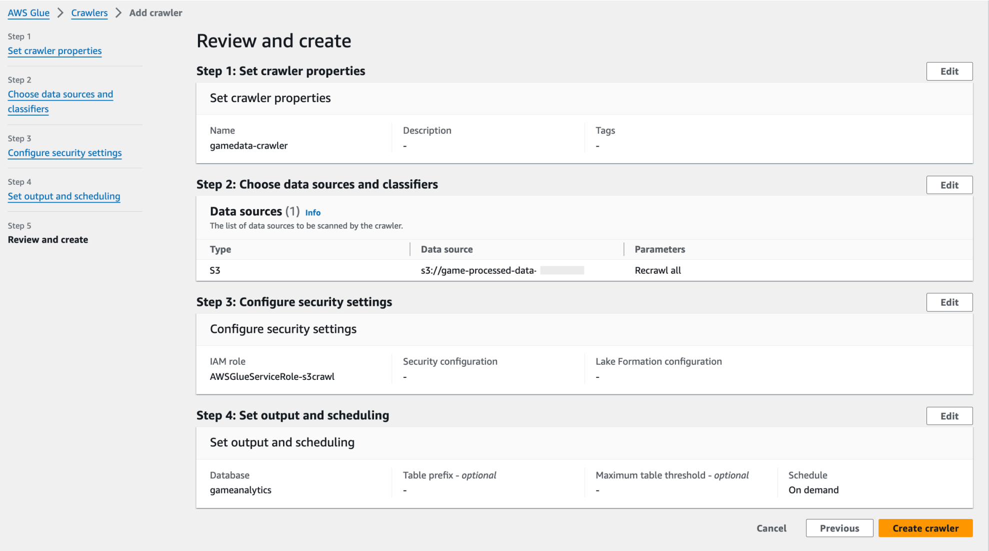
Crawler schedule

You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.

Frequency

On demand ▼

7. Review the crawler and click on **Create crawler**.



AWS Glue > Crawlers > Add crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Review and create

Step 1: Set crawler properties

Set crawler properties

Name	Description	Tags
gamedata-crawler	-	-

Step 2: Choose data sources and classifiers

Data sources (1) Info

The list of data sources to be scanned by the crawler.

Type	Data source	Parameters
S3	s3://game-processed-data-	Recrawl all

Step 3: Configure security settings

Configure security settings

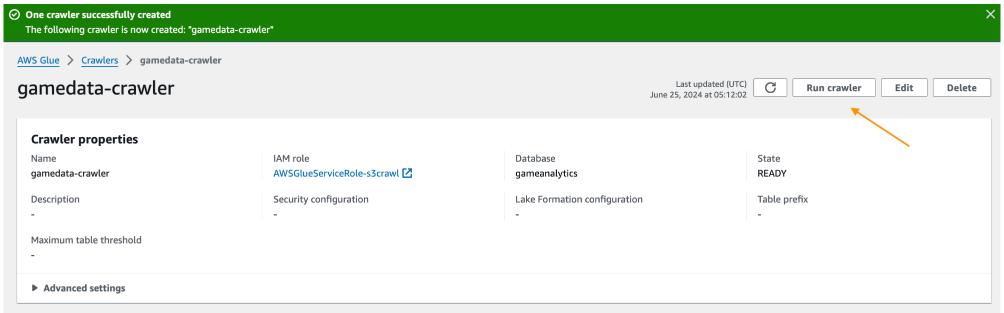
IAM role	Security configuration	Lake Formation configuration
AWSGlueServiceRole-s3crawl	-	-

Step 4: Set output and scheduling

Set output and scheduling

Database	Table prefix - *optional*	Maximum table threshold - *optional*	Schedule
gameanalytics	-	-	On demand

8. After successfully creating the crawler, run the crawler.



One crawler successfully created
The following crawler is now created: "gamedata-crawler"

AWS Glue > Crawlers > gamedata-crawler

gamedata-crawler

Last updated (UTC)
June 25, 2024 at 05:12:02

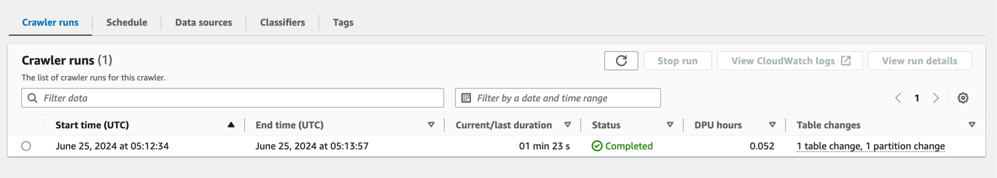
Crawler properties

Name	IAM role	Database	State
gamedata-crawler	AWSGlueServiceRole-s3crawl	gameanalytics	READY

Description	Security configuration	Lake Formation configuration	Table prefix
-	-	-	-

Maximum table threshold
-

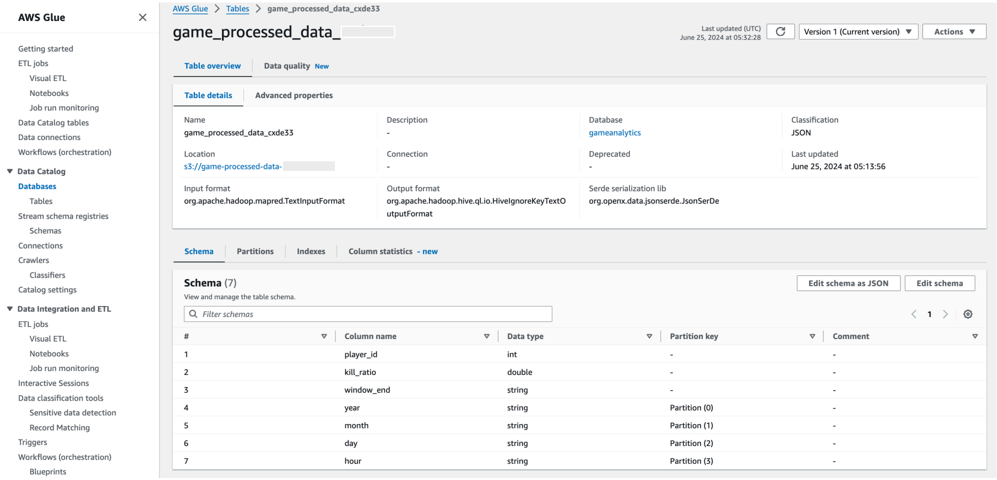
9. Check the status of the crawler.



Crawler runs (1)
The list of crawler runs for this crawler.

	Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes
○	June 25, 2024 at 05:12:34	June 25, 2024 at 05:13:57	01 min 23 s	✓ Completed	0.052	1 table change, 1 partition change

10. Now, you can see the table and schema information under the data catalog section.



AWS Glue > Tables > game_processed_data_cxde33

game_processed_data_

Last updated (UTC)
June 25, 2024 at 05:32:28

Name	Description	Database	Classification
game_processed_data_cxde33	-	gameanalytics	JSON

Location	Connection	Deprecated	Last updated
s3://game-processed-data-	-	-	June 25, 2024 at 05:13:56

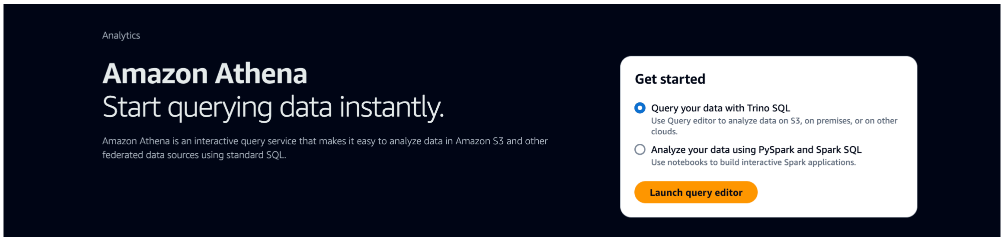
Input format	Output format	Serde serialization lib
org.apache.hadoop.mapred.TextInputFormat	org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat	org.openx.data.jsonserde.JsonSerDe

Schema (7)
View and manage the table schema.

#	Column name	Data type	Partition key	Comment
1	player_id	int	-	-
2	kill_ratio	double	-	-
3	window_end	string	-	-
4	year	string	Partition (0)	-
5	month	string	Partition (1)	-
6	day	string	Partition (2)	-
7	hour	string	Partition (3)	-

Amazon Athena Setup

1. Navigate to the Athena Console and launch query editor.



Analytics

Amazon Athena
Start querying data instantly.

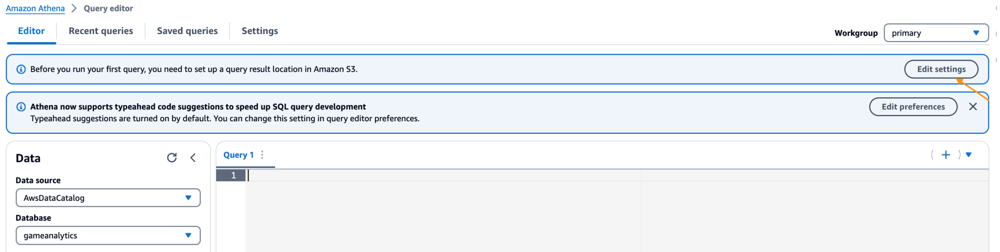
Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 and other federated data sources using standard SQL.

Get started

● Query your data with Trino SQL
Use Query editor to analyze data on S3, on premises, or on other clouds.

○ Analyze your data using PySpark and Spark SQL
Use notebooks to build interactive Spark applications.

2. Click on edit setting to setup query result location in the S3 bucket.



Amazon Athena > Query editor

Workgroup: primary

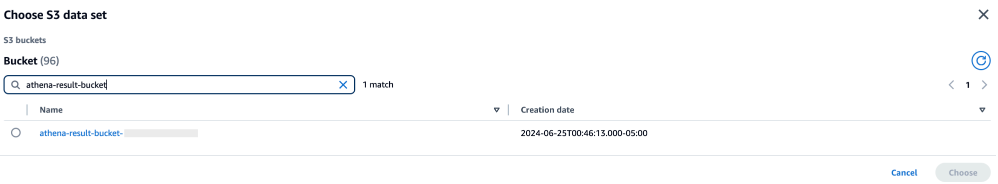
Before you run your first query, you need to set up a query result location in Amazon S3.

Athena now supports typeahead code suggestions to speed up SQL query development
Typeahead suggestions are turned on by default. You can change this setting in query editor preferences.

Data source: AwsDataCatalog

Database: gameanalytics

3. Choose S3 bucket athena-result-bucket-* and **Save** the setting.



Choose S3 data set

S3 buckets

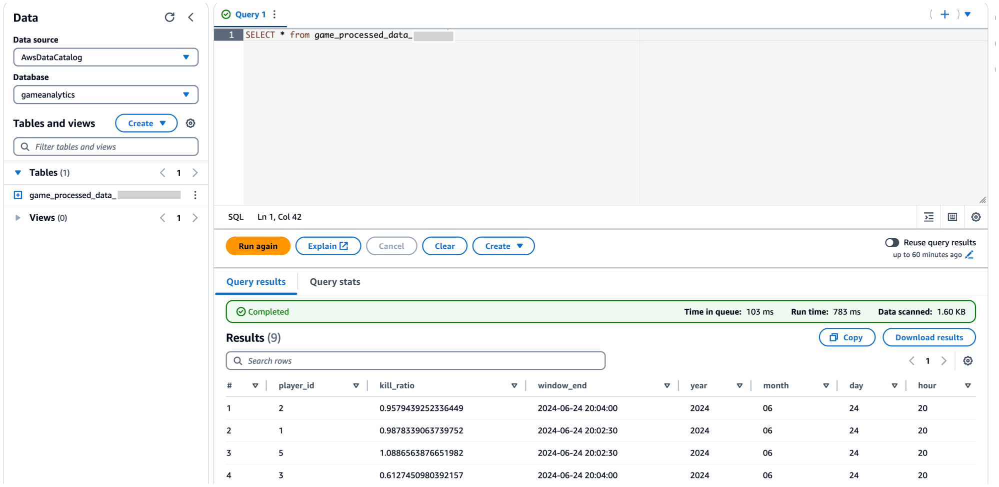
Bucket (96)

athena-result-bucket 1 match

Name	Creation date
○ athena-result-bucket-[redacted]	2024-06-25T00:46:13.000-05:00

4. Choose the **AwsDataCatalog** as the data source, the **gameanalytics** database, and the table name according to your Glue table name. Then, run the following query based on your table name.

```
SELECT * from game_processed_data_xxxxxx;
```



Data

Data source: AwsDataCatalog

Database: gameanalytics

Tables and views

Tables (1)

game_processed_data_[redacted]

Views (0)

Query 1

```
SELECT * from game_processed_data_[redacted]
```

SQL Ln 1, Col 42

Completed Time in queue: 103 ms Run time: 783 ms Data scanned: 1.60 KB

Results (9)

#	player_id	kill_ratio	window_end	year	month	day	hour
1	2	0.9579439252336449	2024-06-24 20:04:00	2024	06	24	20
2	1	0.9878339063739752	2024-06-24 20:02:30	2024	06	24	20
3	5	1.0886563876651982	2024-06-24 20:02:30	2024	06	24	20
4	3	0.6127450980392157	2024-06-24 20:04:00	2024	06	24	20

Conclusion

In this module, you gained hands-on experience managing and delivering processed real-time data to required destinations and making it available for decision making. With this let's move on to next module and visualizing this processed data.

Previous Next