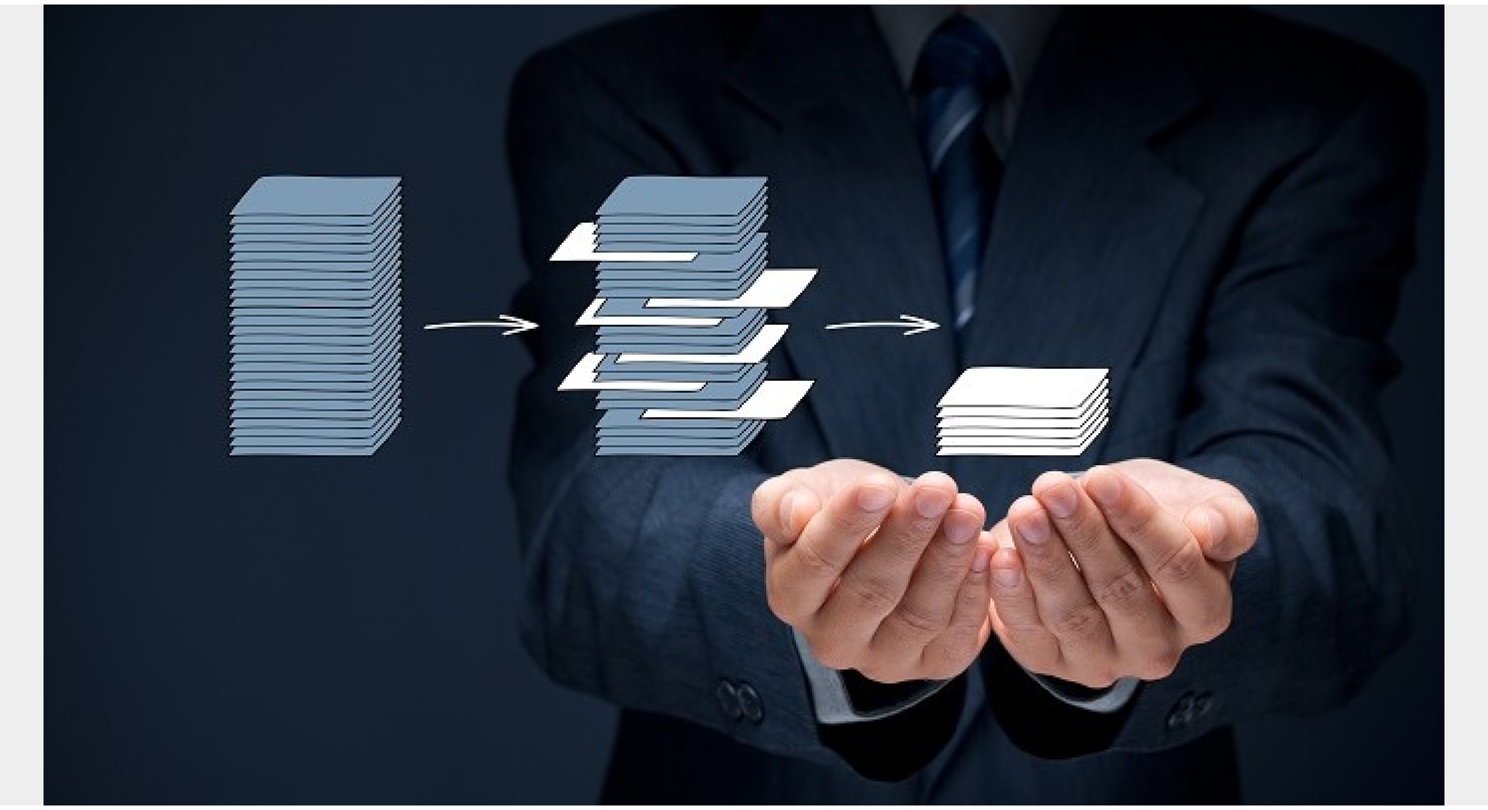
HOW TO MAKE A CRAWLER IN AMAZON GLUE



In this tutorial, we show how to make a crawler in Amazon Glue.

A fully managed service from Amazon, AWS Glue handles data operations like ETL (extract, transform, load) to get the data prepared and loaded for analytics activities. Glue can crawl S3, DynamoDB, and JDBC data sources.

What is a crawler?

A crawler is a job defined in Amazon Glue. It crawls databases and buckets in S3 and then creates tables in Amazon Glue together with their schema.

Then, you can perform your data operations in Glue, like ETL.

Sample data

We need some sample data. Because we want to show how to join data in Glue, we need to have two data sets that have a common element.

The data we use is from IMDB. We have selected a small subset (24 records) of that data and put it into JSON format. (Specifically, they have been formatted to load into DynamoDB, which we will do later.)

One file has the description of a movie or TV series. The other has ratings on that series or movie.

Since the data is in two files, it is necessary to join that data in order to get ratings by title. Glue can do that.

Download these two JSON data files:

- Download title data here.
- Download ratings data here.

```
wget https://raw.githubusercontent.com/werowe/dynamodb/master/100.basics.json
wget
https://raw.githubusercontent.com/werowe/dynamodb/master/100.ratings.tsv.json
```

Upload the data to Amazon S3

Create these buckets in S3 using the Amazon AWS command line client. (Don't forget to run **aws configure** to store your private key and secret on your computer so you can access Amazon AWS.)

Below we create the buckets **titles** and **rating** inside **movieswalker**. The reason for this is Glue will create a separate table schema if we put that data in separate buckets.

(Your top-level bucket name must be unique across all of Amazon. That's an Amazon requirement, since you refer to the bucket by URL. No two customers can have the same URL.)

```
aws s3 mb s3://movieswalker
aws s3 mb s3://movieswalker/titles
aws s3 mb s3://movieswalker/ratings
```

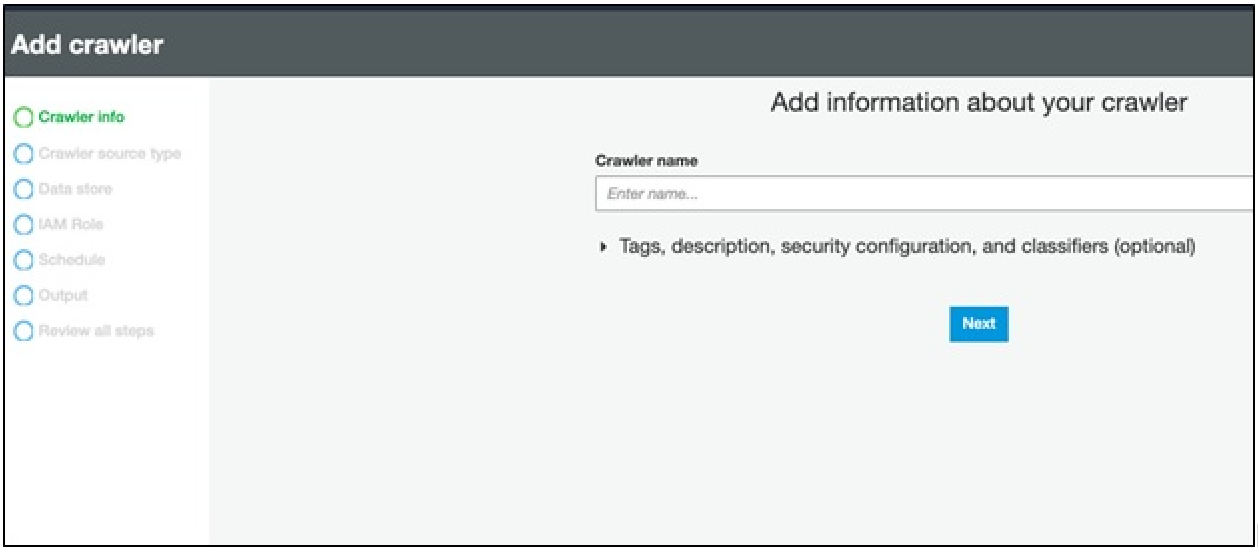
Then copy the title **basics** and **ratings** file to their respective buckets.

```
aws s3 cp 100.basics.json s3://movieswalker/titles
aws s3 cp 100.ratings.tsv.json s3://movieswalker/ratings
```

Configure the crawler in Glue

Log into the Glue console for your AWS region. (Mine is European West.)

Then go to the crawler screen and add a crawler:



Add crawler

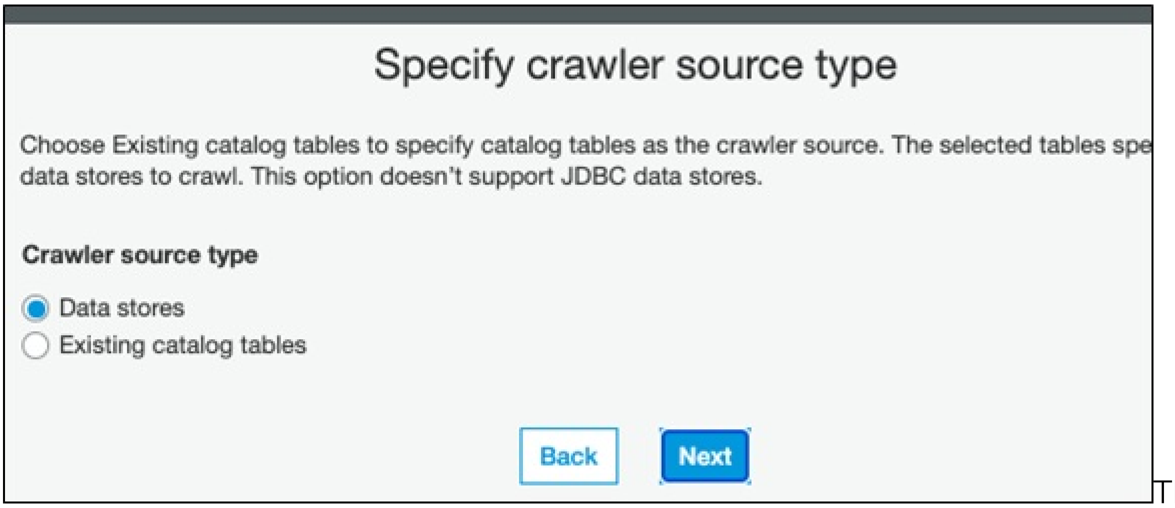
Add information about your crawler

Crawler name

Enter name...

▸ Tags, description, security configuration, and classifiers (optional)

Next, pick a data store. A better name would be **data source**, since we are pulling data from there and storing it in Glue.



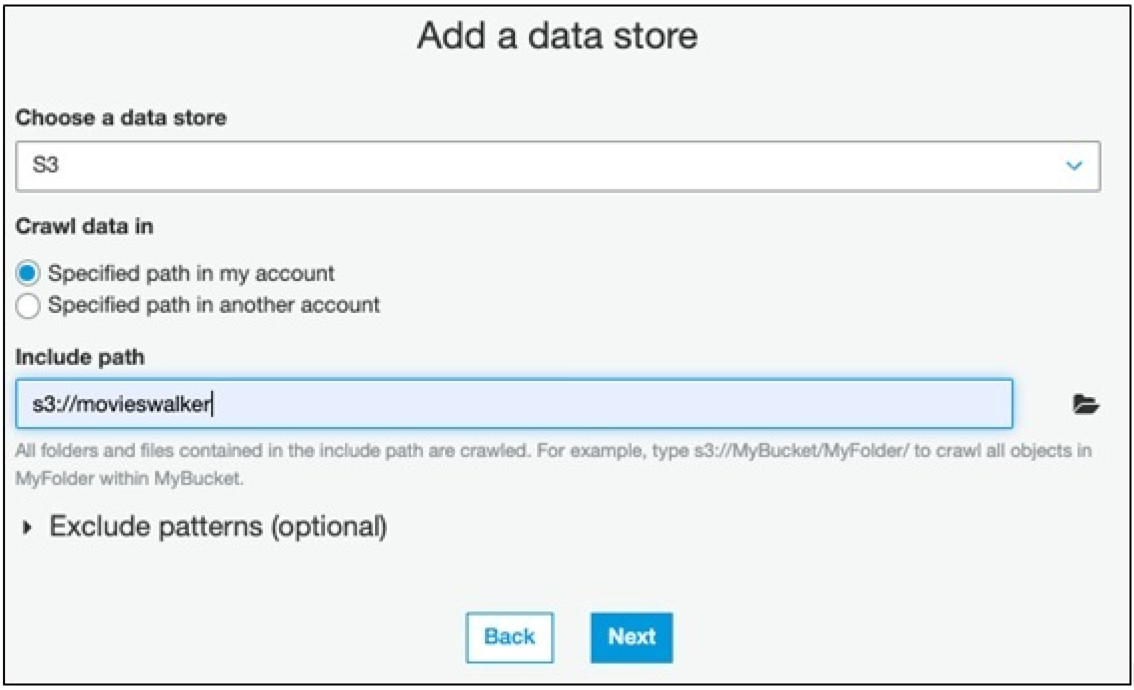
Specify crawler source type

Choose Existing catalog tables to specify catalog tables as the crawler source. The selected tables spe data stores to crawl. This option doesn't support JDBC data stores.

Crawler source type

● Data stores
○ Existing catalog tables

Then pick the top-level movieswalker folder we created above.



Add a data store

Choose a data store

S3

Crawl data in

● Specified path in my account
○ Specified path in another account

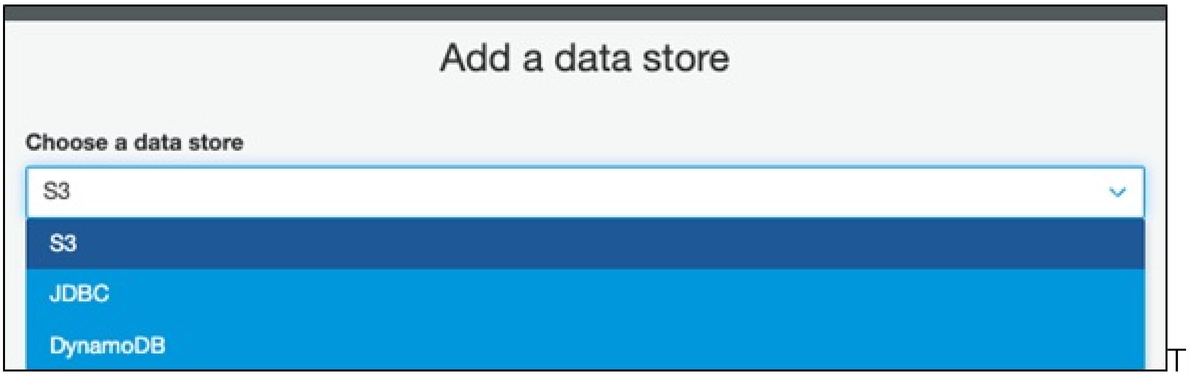
Include path

s3://movieswalker

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

▸ Exclude patterns (optional)

Notice that the data store can be S3, DynamoDB, or JDBC.



Add a data store

Choose a data store

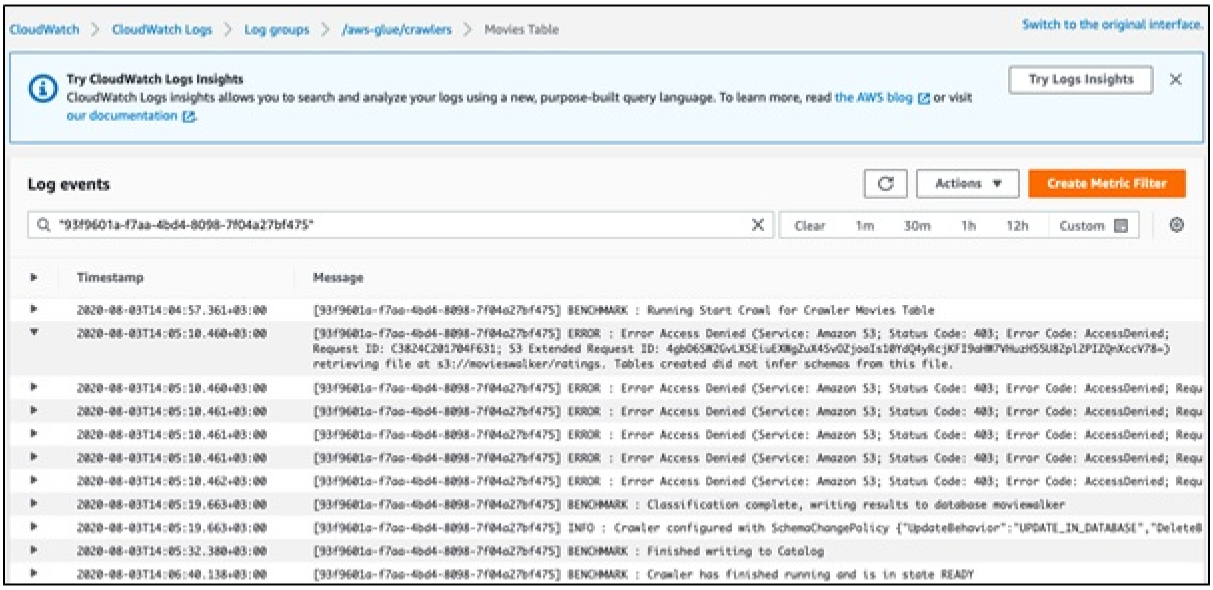
S3

Then start the crawler. When it's done you can look at the logs.

If you get this error it's an S3 policy error. You can make the tables public just for purposes of this tutorial if you don't want to dig into IAM policies. In this case, I got this error because I uploaded the files as the Amazon root user while I tried to access it using a user created with IAM.

```
ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error
Code: AccessDenied; Request ID: 16BA170244C85551; S3 Extended Request ID:
y/JBUpMqsdtf/vnugyFZp8k/DK2cr2hldoXP2JY19NkD39xiTEFp/R8M+UkdO5X1SjrYXuJOnXA=)
retrieving file at s3://movieswalker/100.basics.json. Tables created did not
infer schemas from this file.
```

View the crawler log. Here you can see each step of the process.



CloudWatch > CloudWatch Logs > Log groups > /aws-glue/crawlers > Movies Table
Switch to the original interface.
Try Logs Insights
Log events
Actions
Create Metric Filter
Clear
1m
30m
1h
12h
Custom

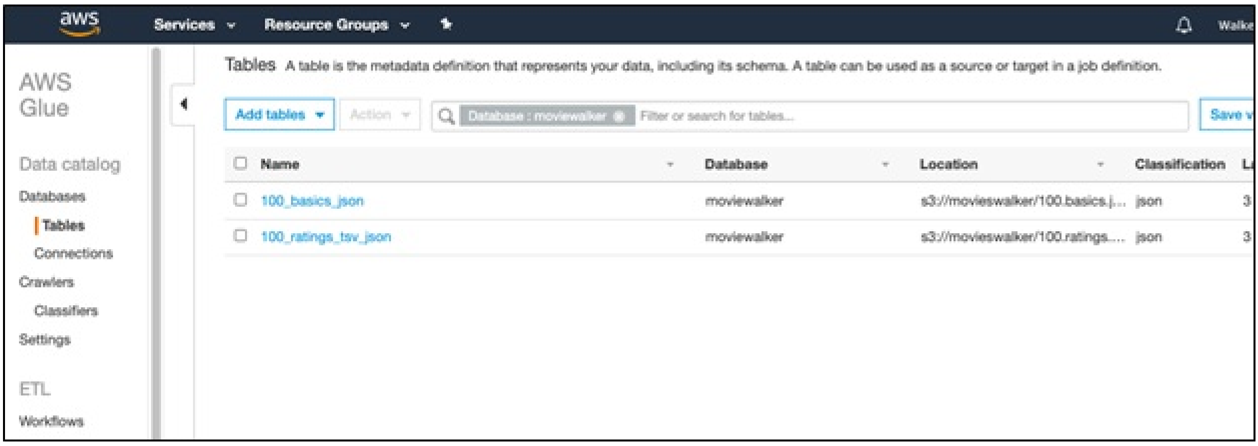
Try CloudWatch Logs Insights
CloudWatch Logs insights allows you to search and analyze your logs using a new, purpose-built query language. To learn more, read the AWS blog or visit our documentation.

"93f9601a-f7aa-4bd4-8098-7f04a27bf475"

```
Timestamp                       Message
2020-08-03T14:04:57.361+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] BENCHMARK : Running Start Crawl for Crawler Movies Table
2020-08-03T14:05:10.460+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied;
                                Request ID: C3824C201704F631; S3 Extended Request ID: 4gbD6SW2GvLXSEiuEXWgZuX4SvOZjooIs10YdQ4yRcjKFI9dHW7VHuzH5SU82pl2PIZQnXccV78=)
                                retrieving file at s3://movieswalker/ratings. Tables created did not infer schemas from this file.
2020-08-03T14:05:10.460+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Requ
2020-08-03T14:05:10.461+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Requ
2020-08-03T14:05:10.461+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Requ
2020-08-03T14:05:10.461+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Requ
2020-08-03T14:05:10.462+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] ERROR : Error Access Denied (Service: Amazon S3; Status Code: 403; Error Code: AccessDenied; Requ
2020-08-03T14:05:19.663+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] BENCHMARK : Classification complete, writing results to database moviewalker
2020-08-03T14:05:19.663+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] INFO : Crawler configured with SchemaChangePolicy {"UpdateBehavior":"UPDATE_IN_DATABASE","DeleteB
2020-08-03T14:05:32.380+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] BENCHMARK : Finished writing to Catalog
2020-08-03T14:06:40.138+03:00   [93f9601a-f7aa-4bd4-8098-7f04a27bf475] BENCHMARK : Crawler has finished running and is in state READY
```

View tables created in Glue

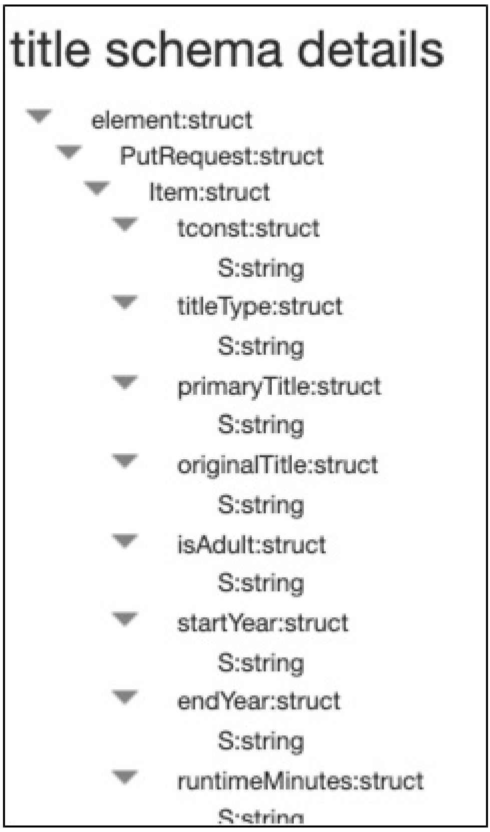
Here are the tables created in Glue.



Tables A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

□ Name	Database	Location	Classification	L
□ 100_basics_json	moviewalker	s3://movieswalker/100.basics.j...	json	3
□ 100_ratings_tsv_json	moviewalker	s3://movieswalker/100.ratings....	json	3

If you click on them you can see the schema.



title schema details

```
element:struct
  PutRequest:struct
    Item:struct
      tconst:struct
        S:string
      titleType:struct
        S:string
      primaryTitle:struct
        S:string
      originalTitle:struct
        S:string
      isAdult:struct
        S:string
      startYear:struct
        S:string
      endYear:struct
        S:string
      runtimeMinutes:struct
        S:string
```

It has these properties. The item of interest to note here is it stored the data in Hive format, meaning it must be using Hadoop to store that.

```
{
        "StorageDescriptor": {
                "cols": {
                        "FieldSchema":
                },
                "location": "s3://movieswalker/100.basics.json",
                "inputFormat": "org.apache.hadoop.mapred.TextInputFormat",
                "outputFormat":
"org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat",
                "compressed": "false",
                "numBuckets": "-1",
                "SerDeInfo": {
                        "name": "",
                        "serializationLib":
```

```
"org.openx.data.jsonserde.JsonSerDe",
                        "parameters": {
                                "paths": "title"
                        }
                },
                "bucketCols": [],
                "sortCols": [],
                "parameters": {
                        "sizeKey": "7120",
                        "UPDATED_BY_CRAWLER": "S3 Movies",
                        "CrawlerSchemaSerializerVersion": "1.0",
                        "recordCount": "1",
                        "averageRecordSize": "7120",
                        "CrawlerSchemaDeserializerVersion": "1.0",
                        "compressionType": "none",
                        "classification": "json",
                        "typeOfData": "file"
                },
                "SkewedInfo": {},
                "storedAsSubDirectories": "false"
        },
        "parameters": {
                "sizeKey": "7120",
                "UPDATED_BY_CRAWLER": "S3 Movies",
                "CrawlerSchemaSerializerVersion": "1.0",
                "recordCount": "1",
                "averageRecordSize": "7120",
                "CrawlerSchemaDeserializerVersion": "1.0",
                "compressionType": "none",
                "classification": "json",
                "typeOfData": "file"
        }
}
```

Additional resources

For more on this topic, explore these resources:

- BMC Machine Learning & Big Data Blog
- BMC Multi-Cloud Blog
- Our multi-part AWS Guide
- Is ETL (Extract, Transform, Load) Still Relevant?
- Using Python for Big Data and Analytics
- Simplifying and Scaling Data Pipelines in the Cloud
- Structured vs Unstructured Data: A Shift in Privacy