Amazon Web Services
Data Engineering Immersion Day

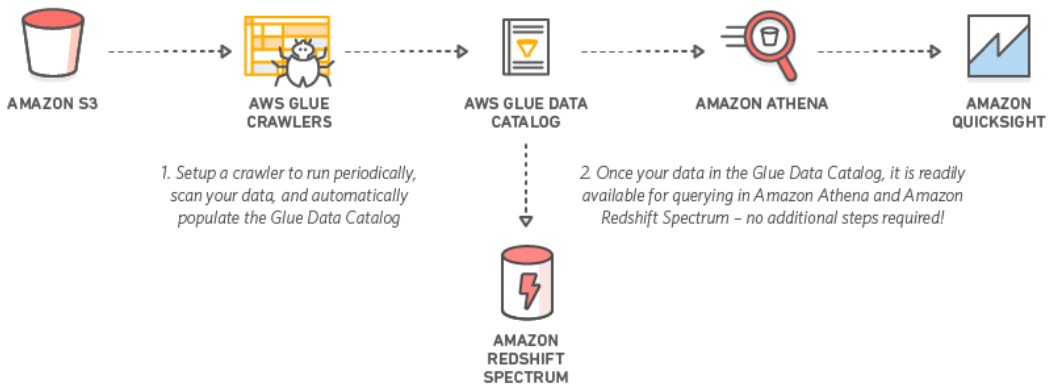
Extract, Transform and Load Data Lake with Glue

Jun 2019

Table of Contents

Introduction

This lab will give you an understanding of the AWS Glue – a fully managed data catalog and ETL service, as well as Athena and Quicksight for querying and visualization the data you import.



Prerequisites:

The DMS Lab is a prerequisite for this lab.

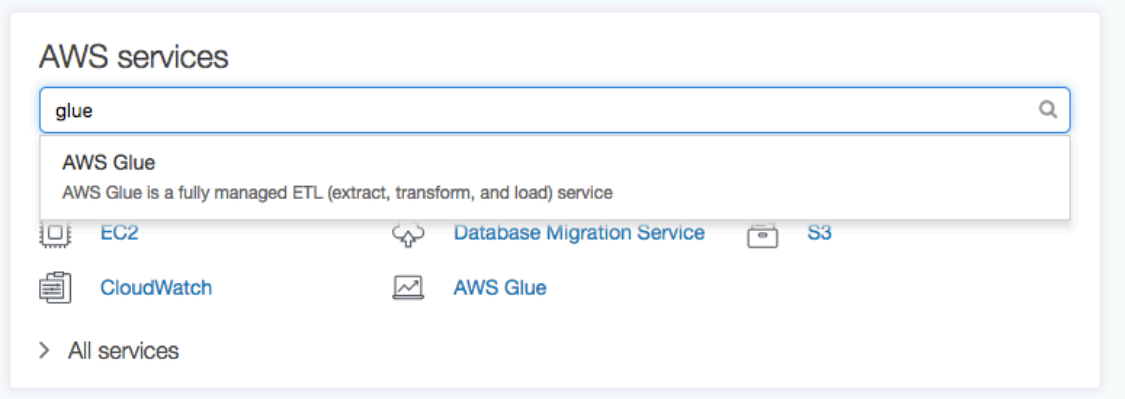
Tasks Completed in this Lab:

In this lab you will be completing the following tasks:

1. Create Glue crawler for initial data
2. Create Glue crawler for ongoing replication (optional)
3. Create Glue ETL to transform CSV data to Parquet format

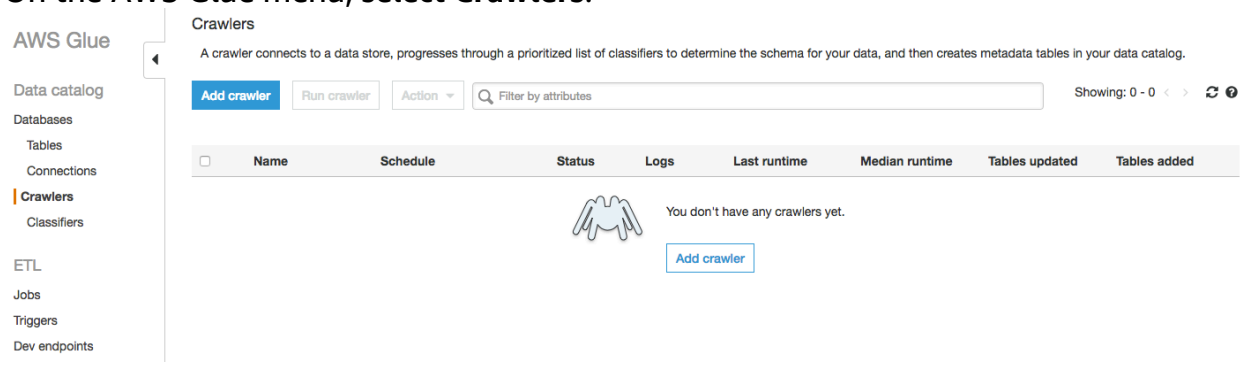
Getting Started

Navigate to the AWS Glue service.

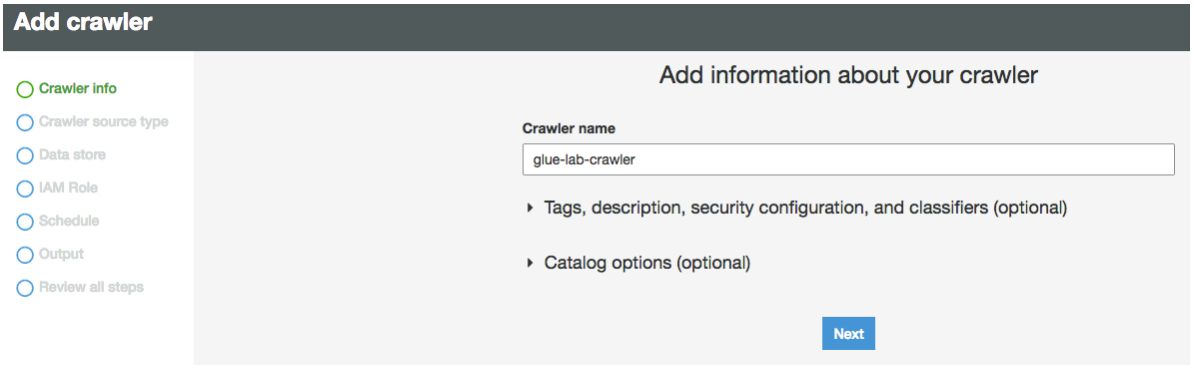


Create Glue Crawler for initial full load data

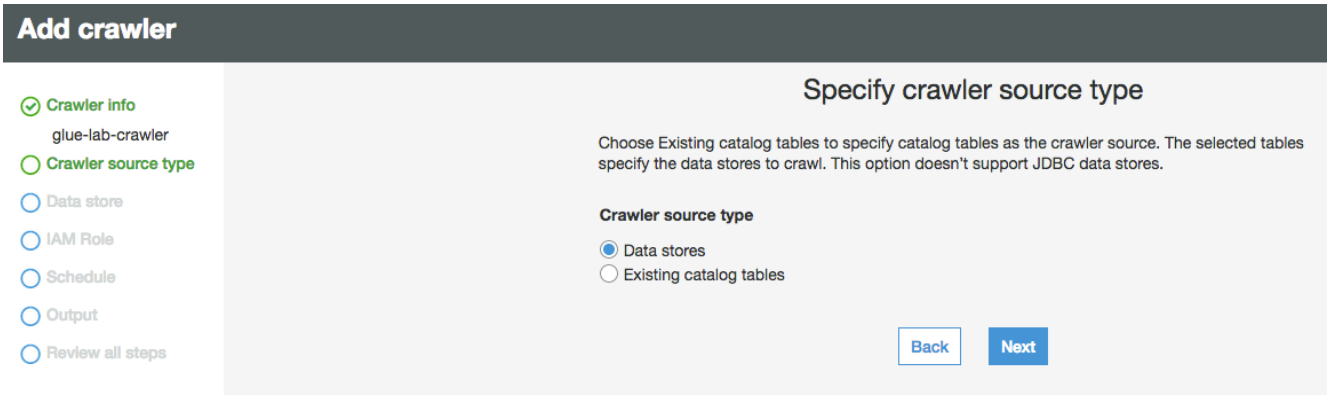
1. On the AWS Glue menu, select **Crawlers**.



2. Click **Add crawler**.
3. Enter the crawler name for initial data load. This name should be descriptive and easily recognized (e.g ," glue-lab-crawler").
4. Optionally, enter the description. This should also be descriptive and easily recognized and Click **Next**.



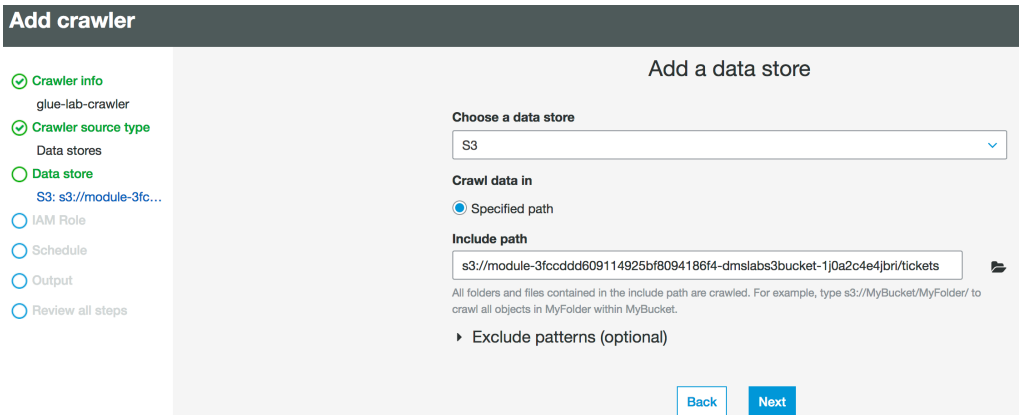
5. Choose **Crawler Source Type** as **Data Source** and **Click Next**



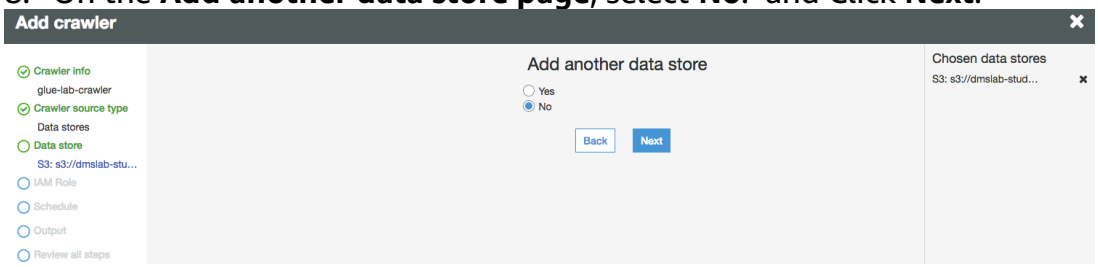
6. On the **Add a data store** page, make the following selections:

a. For Choose a data store, click the drop-down box and select **S3**.
b. For Crawl data in, select **Specified path in my account**.
c. For **Include** path, click the little folder icon, browse to the tickets folder e.g., "s3://xxxxx/**tickets**

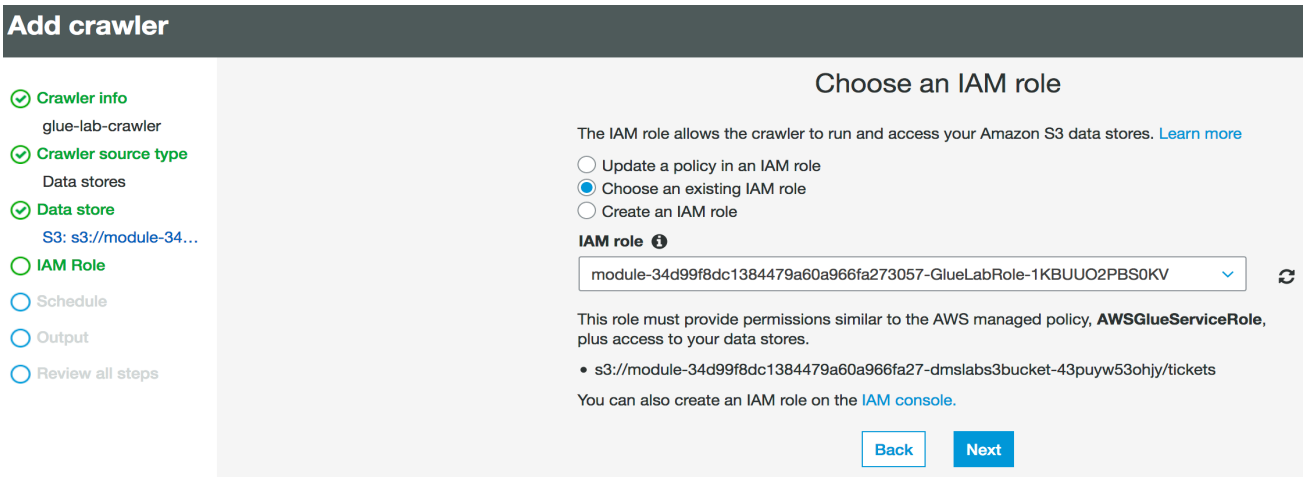
7. Click **Next**.



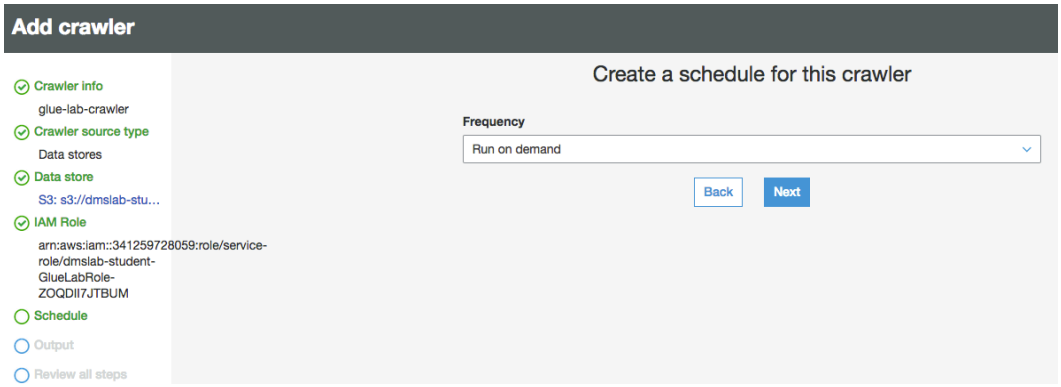
8. On the **Add another data store page**, select **No**. and Click **Next**.



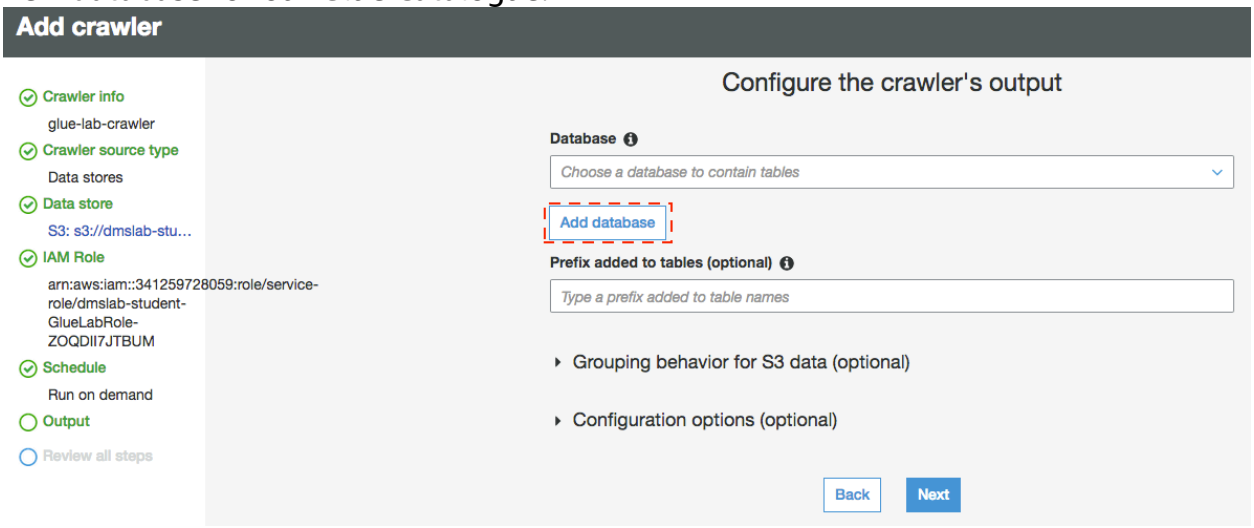
9. On the **Choose an IAM role** page, make the following selections:
   a. Select **Choose an existing IAM role**.
   b. **For IAM role**, select the **GlueLabRole** that was created at the initial environment setup
10. Click **Next**.



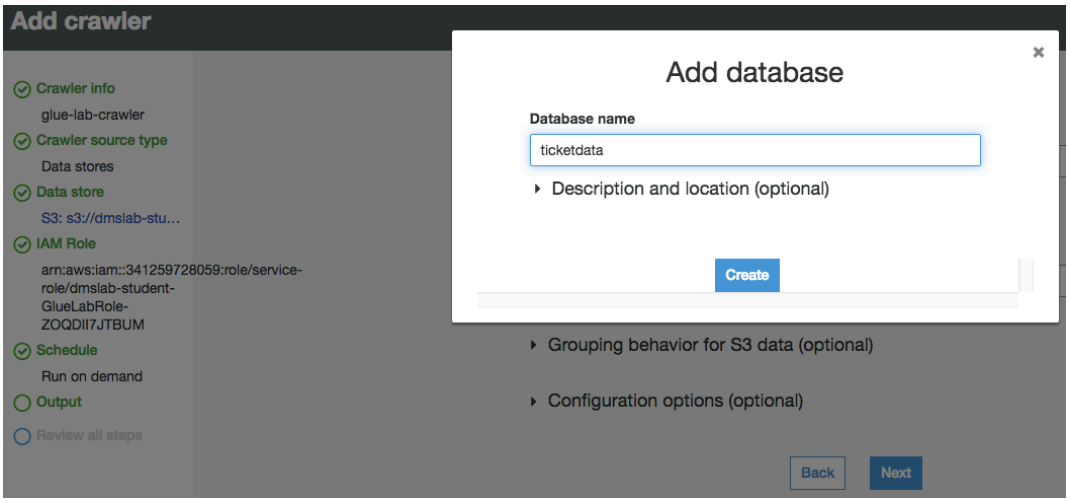
11. On the Create a schedule for this crawler page, for Frequency, select **Run on demand** and Click **Next**.



12. On the Configure the crawler's output page, click **Add database** to create a new database for our Glue Catalogue.

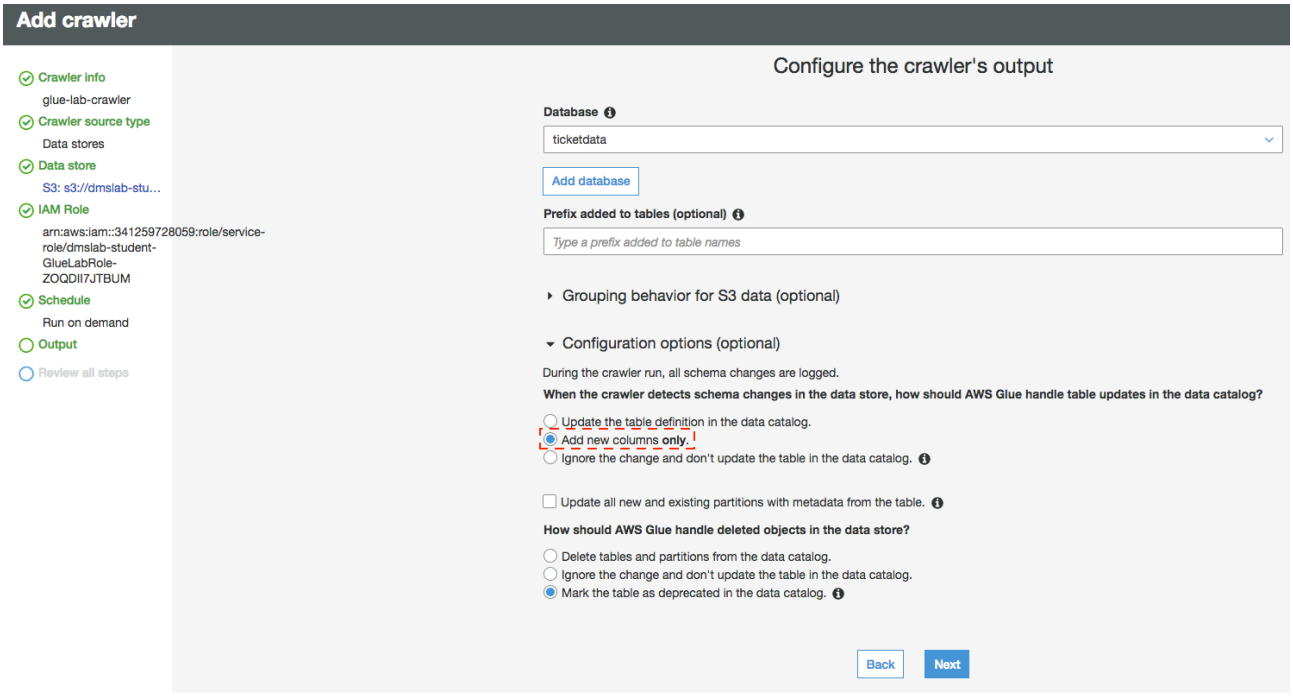


13. Give Catalog database name as per your convenient choice for example "**ticktdata**" and click **create**

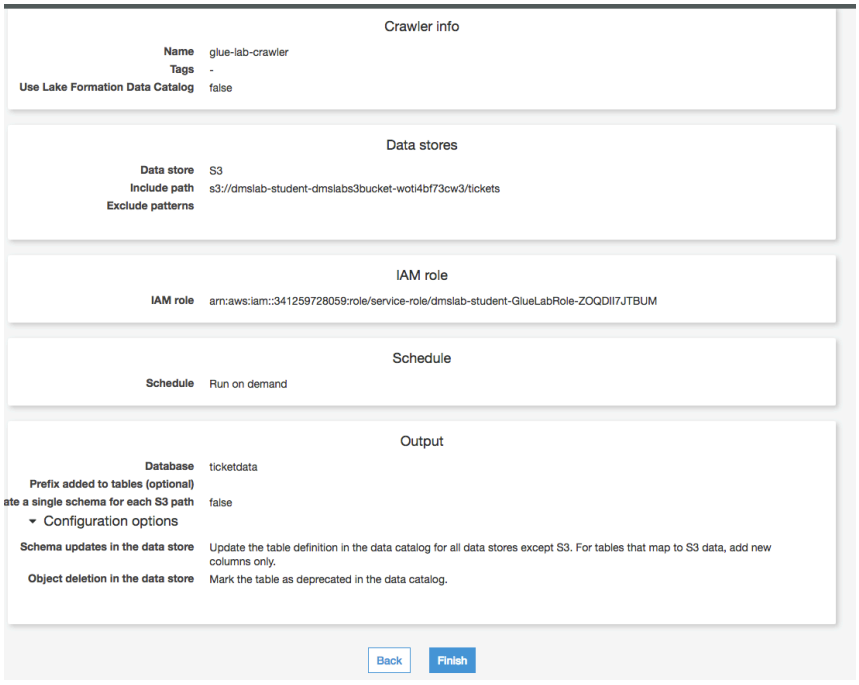


14. For Prefix added to tables (optional), leave the field empty.

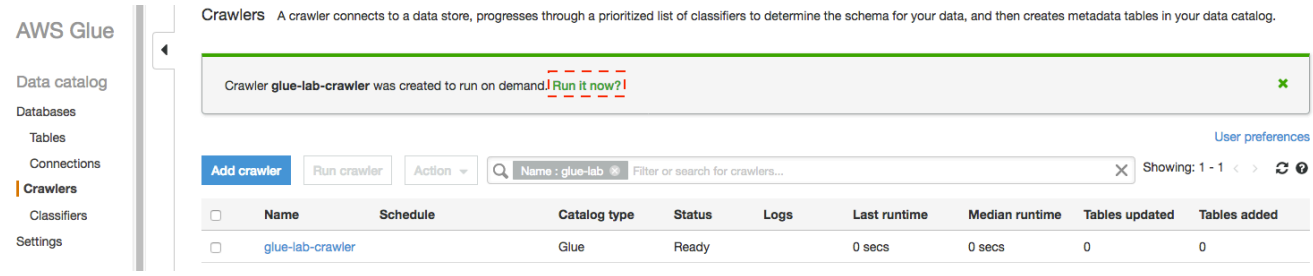
15. For Configuration options (optional), select **Add new columns only** and keep the remaining default configuration options and Click **Next**.



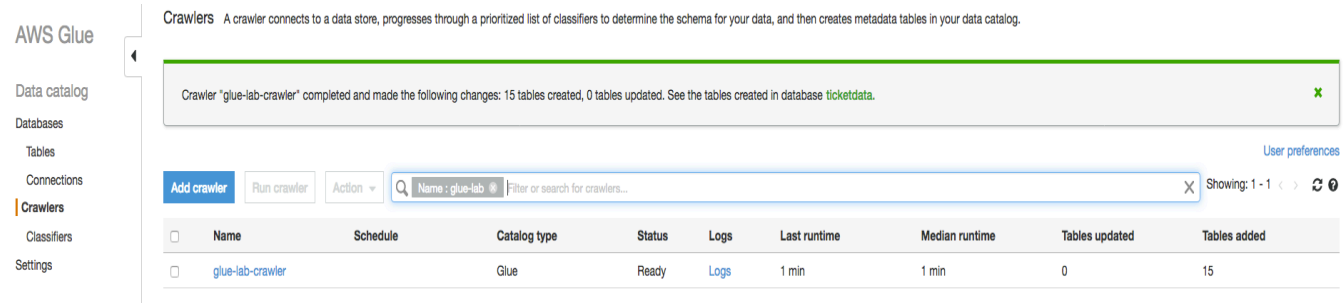
16. Review the summary page noting the Include path and Database output and Click **Finish**. The crawler is now ready to run.



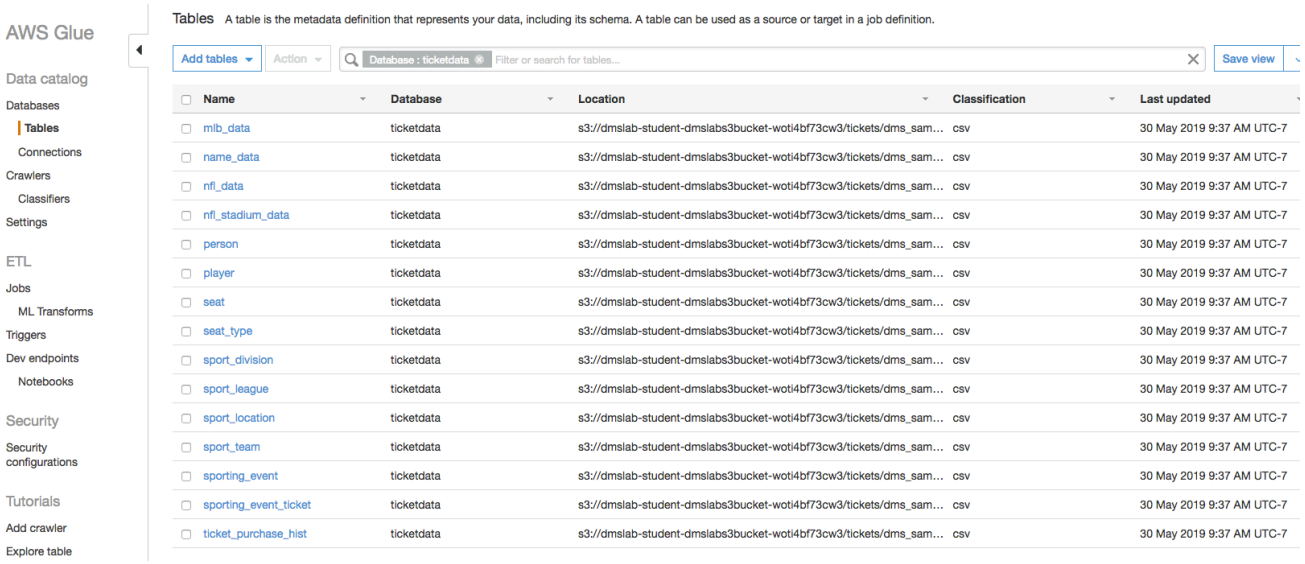
17. Click **Run it now**.



Crawler will change status from starting to stopping, wait until crawler comes back to ready state, you can see that it has created 15 tables.

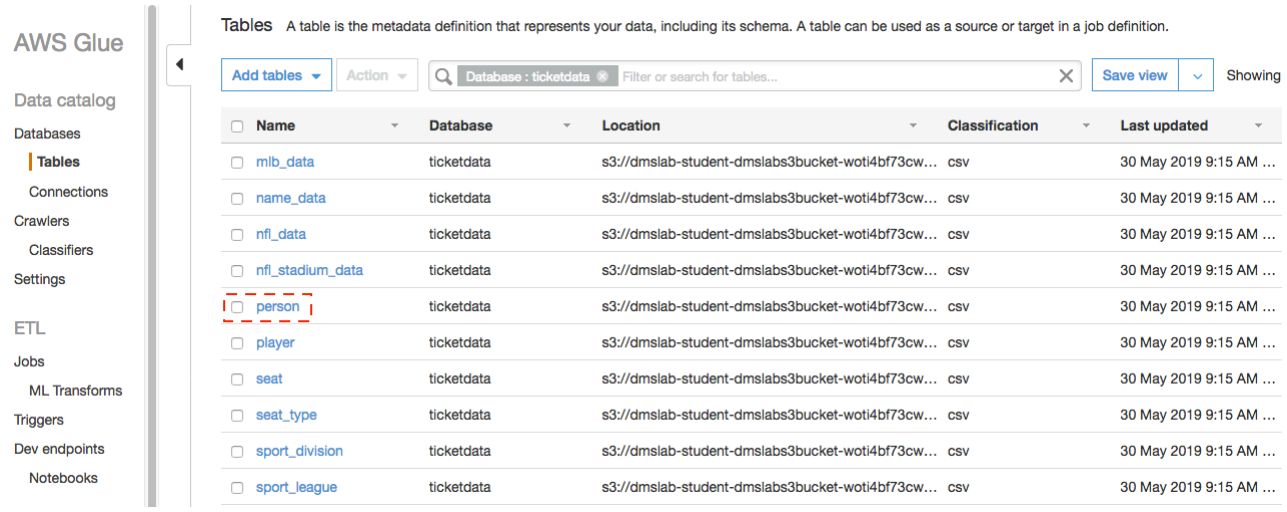


18. In the AWS Glue navigation pane, click **Databases** > **Tables**. (You can also click the database name (e.g., "ticketdata" to browse the tables.).



Data Validation Exercise

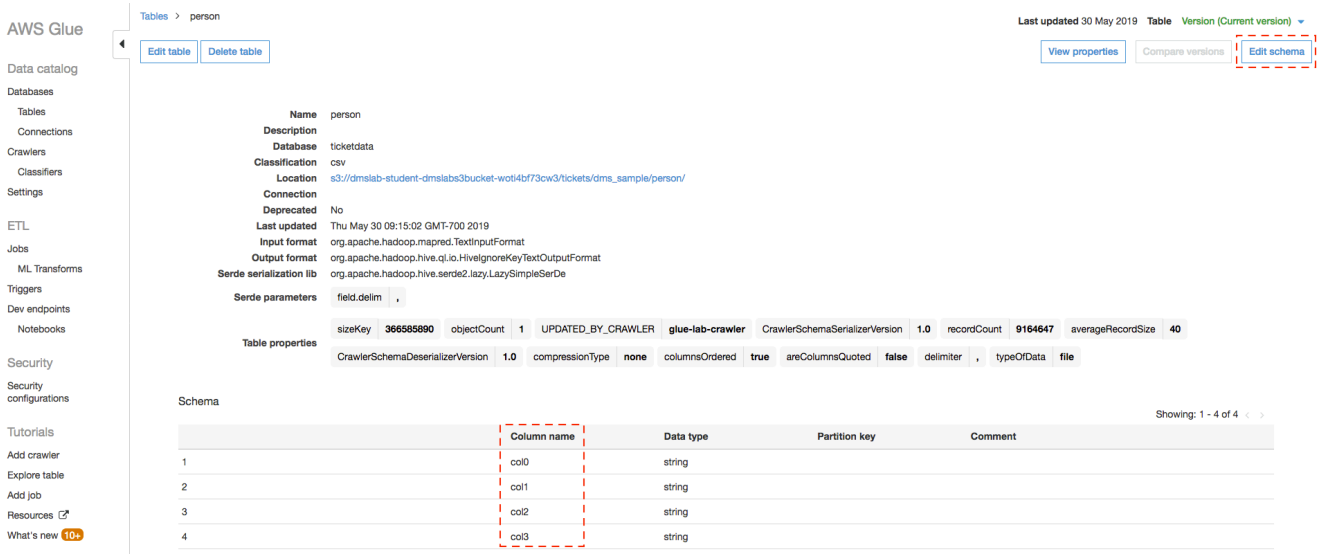
1. Within the Tables section of your ticketdata database, click the person table.



You may have noticed that some tables (such as person) have column headers such as **col0,col1,col2,col3**. In absence of headers or when the crawler cannot determine the header type, default column headers are specified.

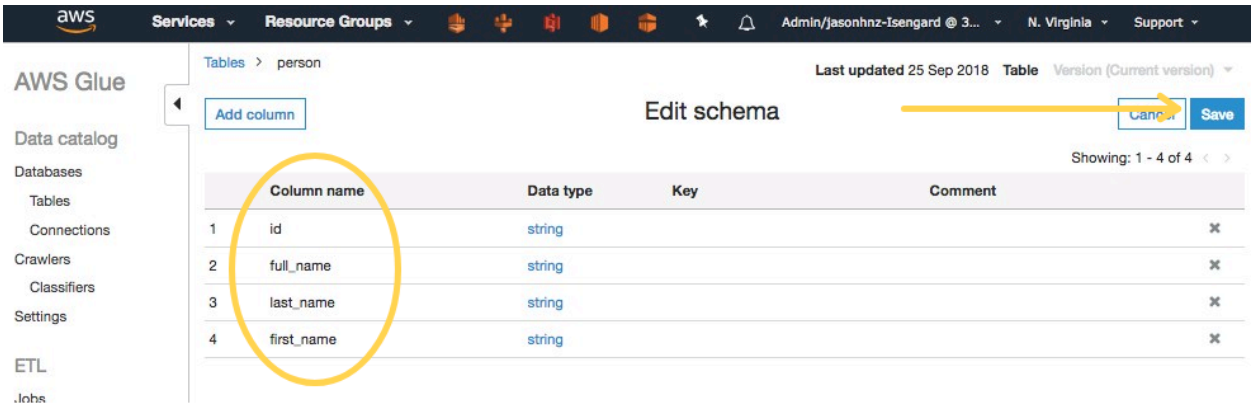
This exercise uses the person table as an example of how to resolve this issue.

2. Click **Edit Schema** on the top right side.



3. In the Edit Schema section, double-click **col0** (column name) to open edit mode. Type “id” as the column name.

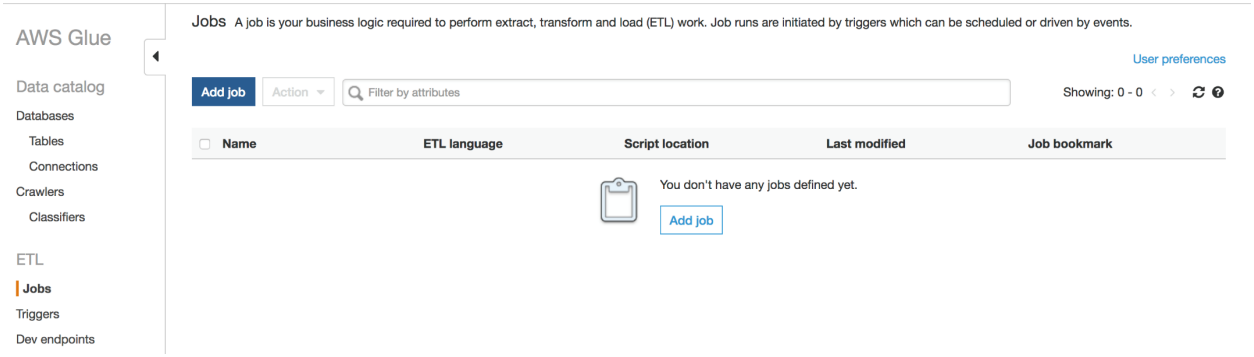
4. Repeat the preceding step to change the remaining column names to match those shown in the following figure.



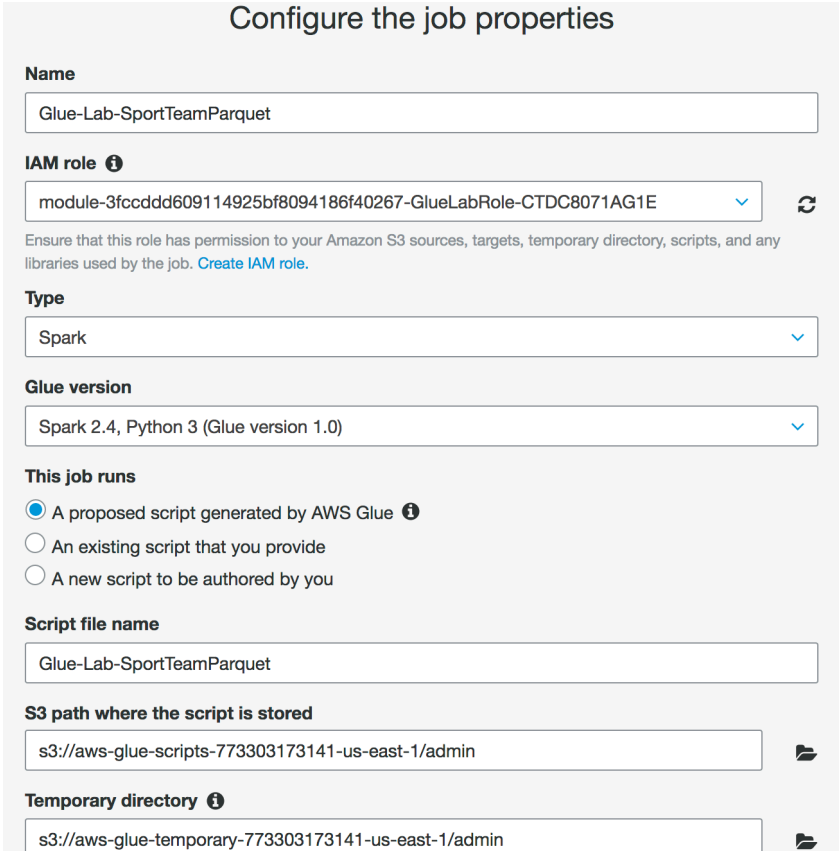
5. Click **Save**.

Data ETL Exercise

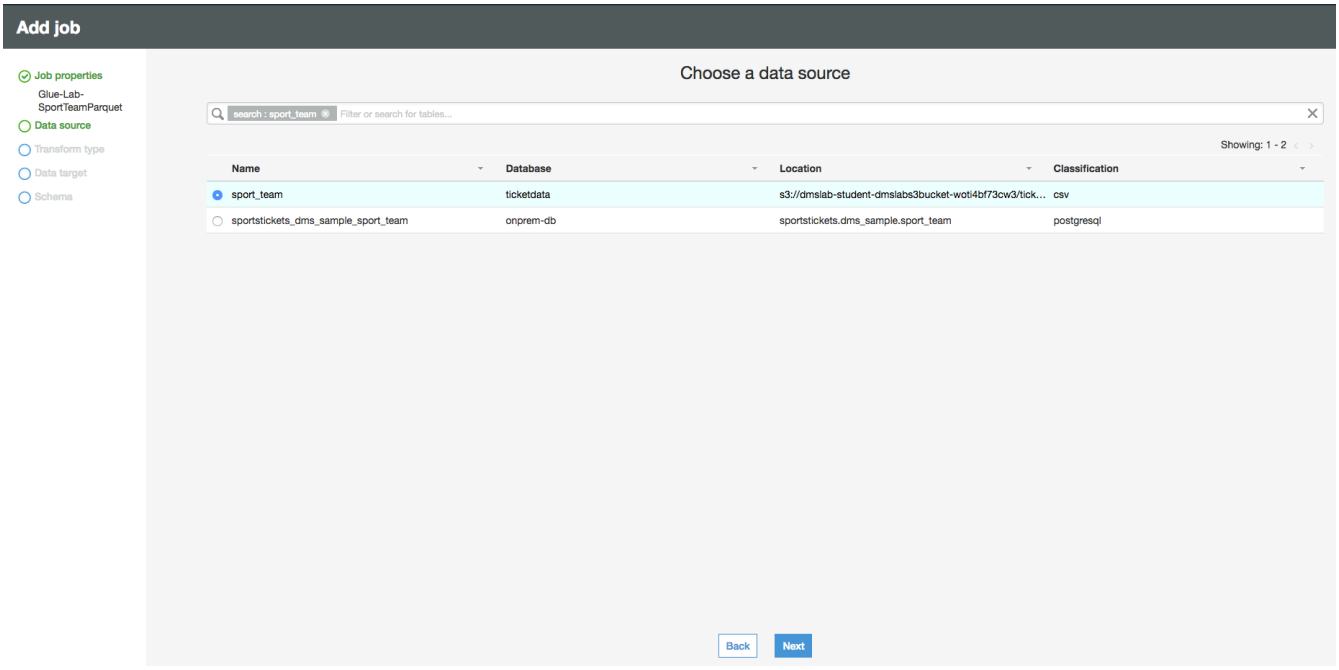
1. In the left navigation pane, under **ETL**, click **Jobs**, and then click **Add job**.



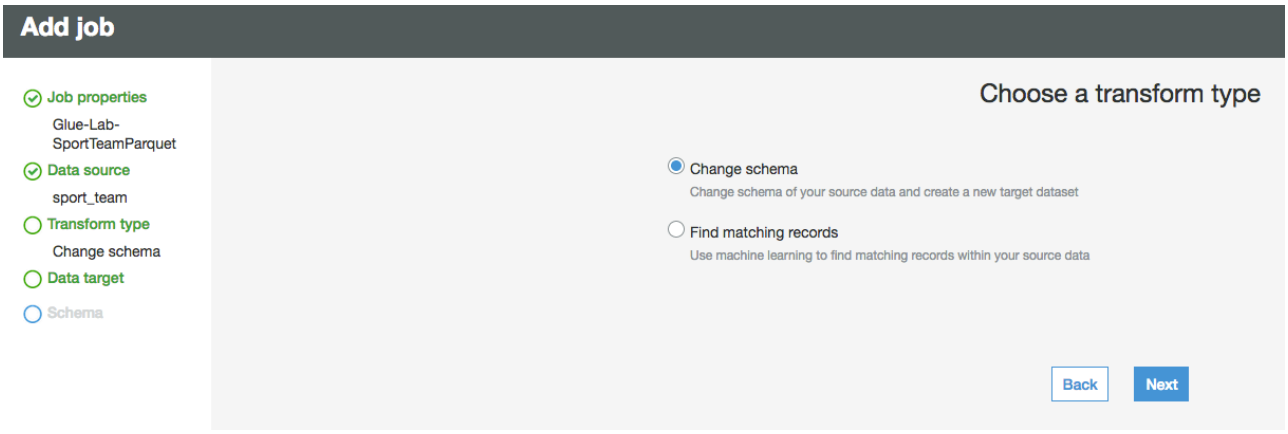
2. On the Job properties page, make the following selections:
   a. For **Name**, type **Glue-Lab-SportTeamParquet**.
   b. For **IAM role**, choose existing **GlueLabRole**
   c. For **Type**, Select **Spark**
   d. Choose **Python 3** in Glue Version
   e. For **This job runs**, select **A proposed script generated by AWS Glue**.
   f. For Script file name, type **Glue-Lab-SportTeamParquet**.
   g. Keep the rest settings as **default**.
3. Click **Next**.



4. On the **Data source** page, select **sport_team** and Click **Next**.

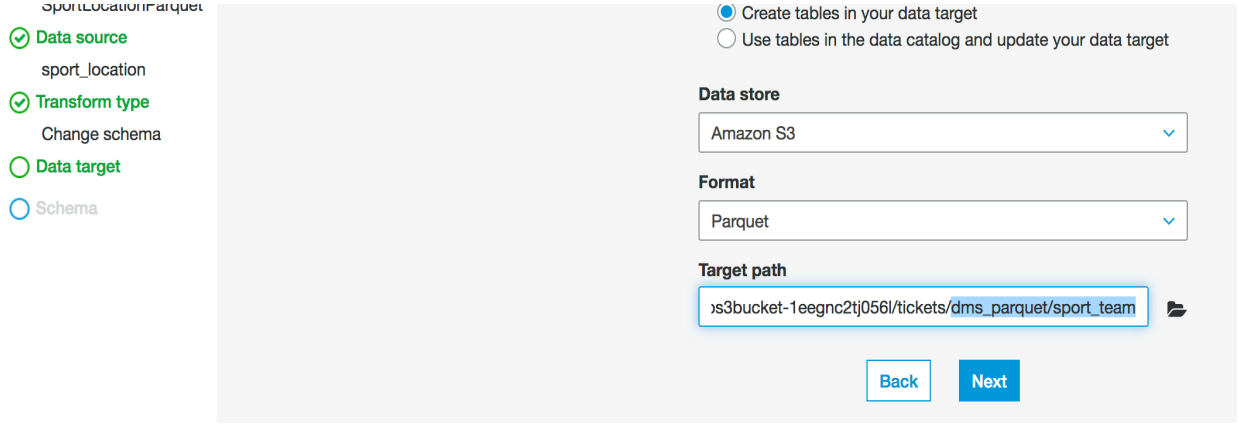


5. On the **Choose a transformation type** page, select **change schema**

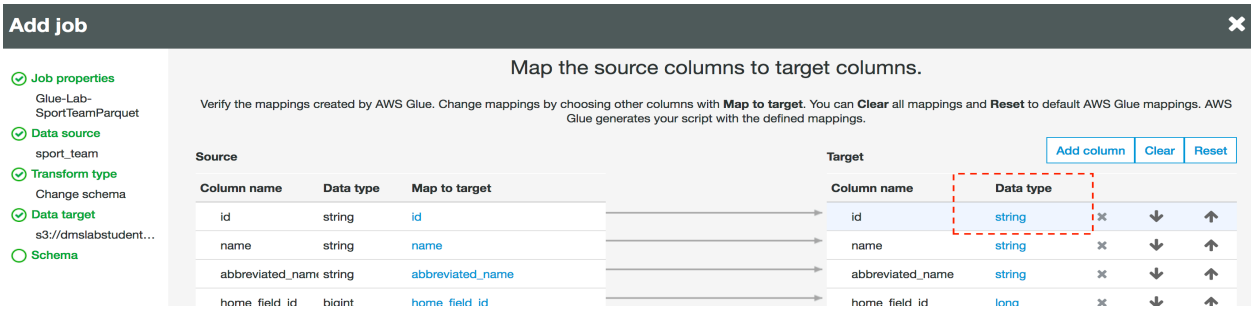


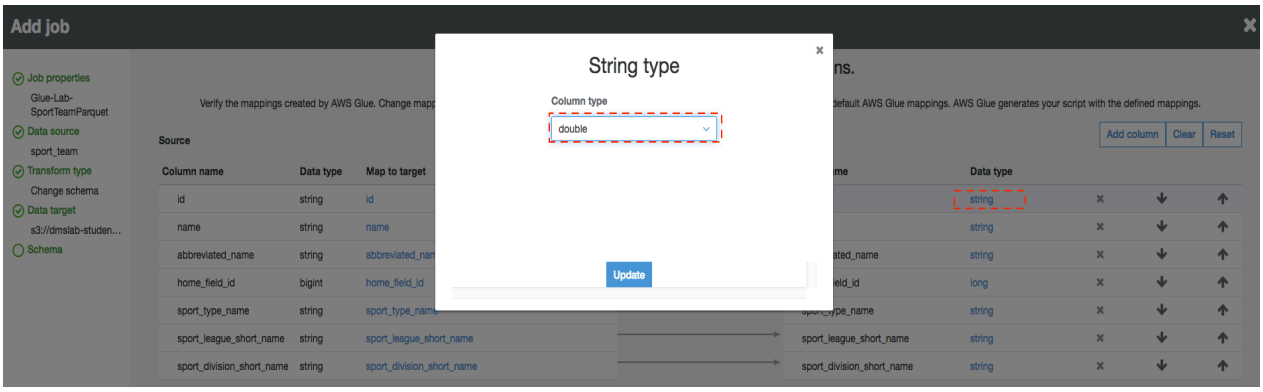
6. On the **Choose a data target** page, select **Create tables in your data target**.
7. For Data store, select **Amazon S3**.
8. For Format, select **Parquet**.
9. For Target path, create a *new folder* **dms_parquet** for the table **sport_team** at the end of the path, ie. s3://<bucketname>/tickets/**dms_parquet/sport_team** (an empty folder), to store the results produced by the ETL Job *Glue-Lab-SportTeamParquet*

10. Click **Next**.



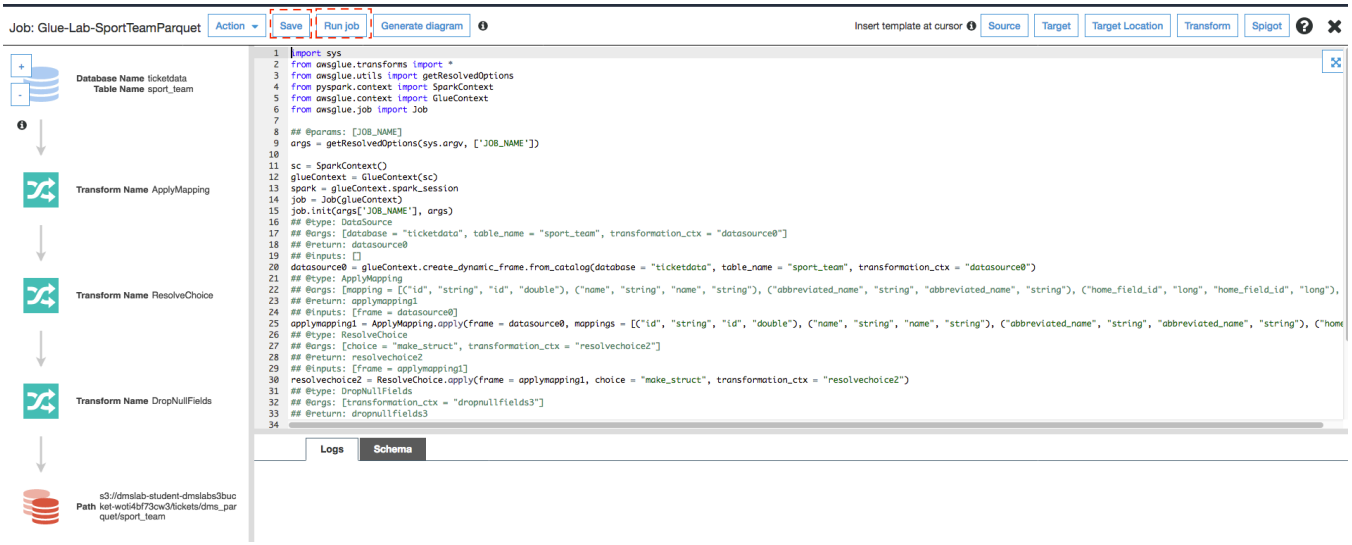
11. Click the **target Data type** to edit the **id** schema mapping. In **String type** pop-up window Select **double** from **Column type** drop down and click **update**.



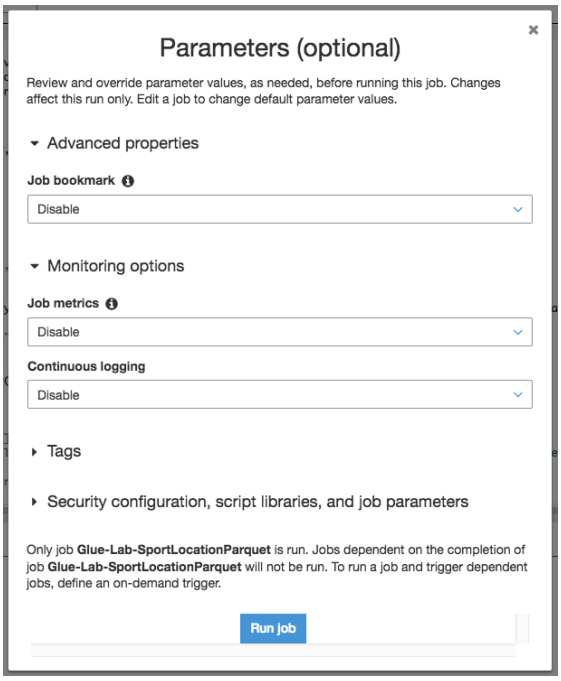


12. click **Save job and edit script**.

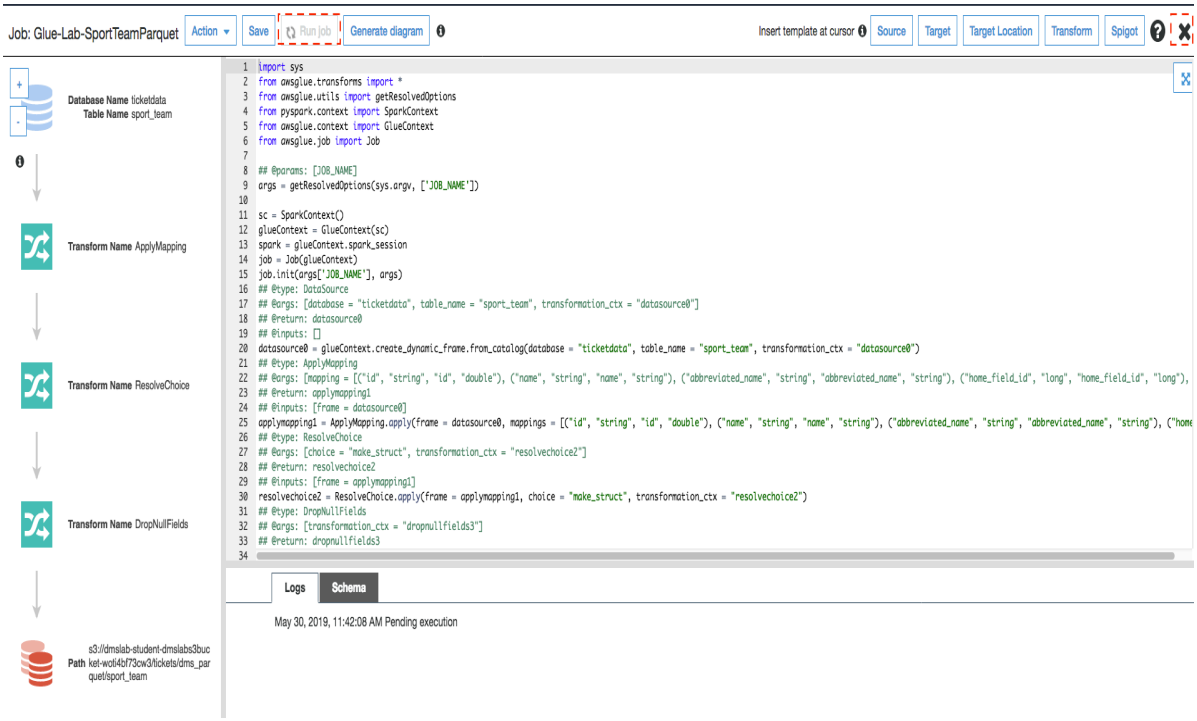
13. View the job. (This screen provides you with the ability to customize this script as required.) Click **Save** and then **Run Job**.



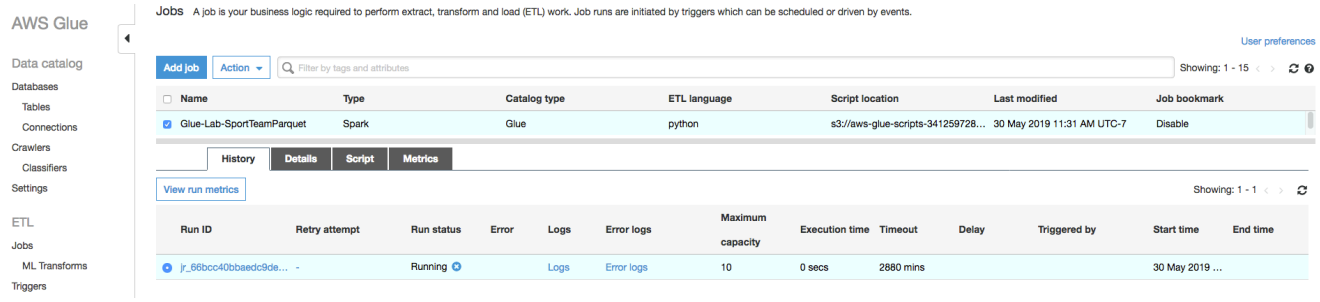
14. In **Parameters** option,
    a. you can leave **Job bookmark** as **Disable**. AWS Glue tracks data that has already been processed during a previous run of an ETL job by persisting state information from the job run.
    b. You can leave the **Job metrics** option **Disable**. You can collect metrics about AWS Glue jobs and visualize them on the AWS Glue with job metrics.



15. Click **Run Job**
16. You will see job is now running as **Run job** button got disable. Click the cross button located on the top right corner to close the window to return to the ETL jobs.



17. Click your job to view history and verify that it ran successfully.



IMPORTANT: We will **repeat** the preceding steps to create **4 more new ETL Jobs** to transform the additional tables from CSV to Parquet format.

#	Job Name & Script Filename	Source Table	S3 Target Path
1	Glue-Lab-SportLocationParquet	sport_location	dms_parquet/sport_location
2	Glue-Lab-SportingEventParquet	sporting_event	dms_parquet/sporting_event ***(require 2 data type change, see below)***
3	Glue-Lab-SportingEventTicketParquet	sporting_event_ticket	dms_parquet/sporting_event_ticket ***(require 3 data type change, see below)***
4	Glue-Lab-PersonParquet	person	dms_parquet/person ***(require 1 data type change, see below)***

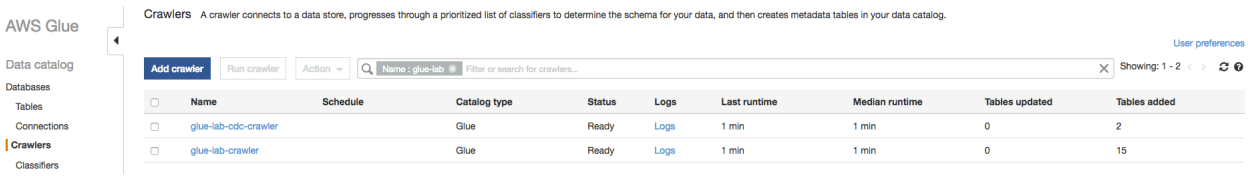
To enable us to join these tables, we will also update the **target data types** in the schema.

#	Table	Column	Source Data Type	Target Data Type
1	sporting_event	start_date_time	STRING	TIMESTAMP
2	sporting_event	start_date	STRING	DATE
3	sporting_event_ticket	id	STRING	DOUBLE
4	sporting_event_ticket	sporting_event_id	STRING	DOUBLE
5	sporting_event_ticket	tickerholder_id	STRING	DOUBLE
6	person	id	STRING	DOUBLE

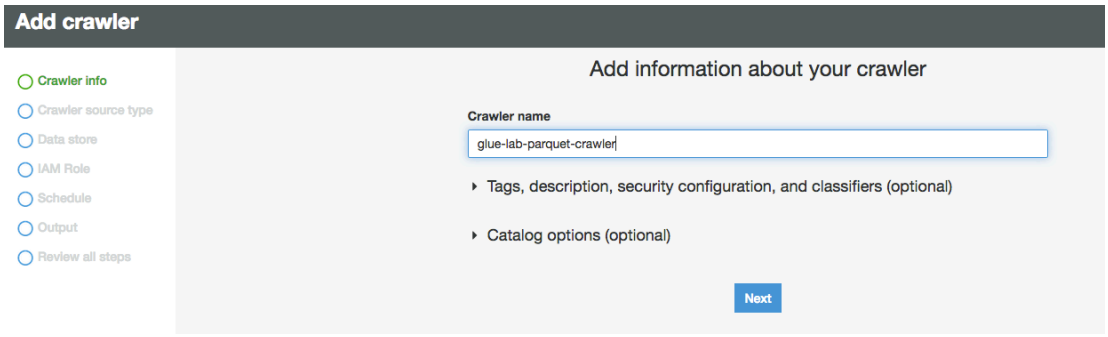
Once these jobs have completed, we can create a crawler to detect metadata information for these parquet files.

Create Crawler for Parquet Files

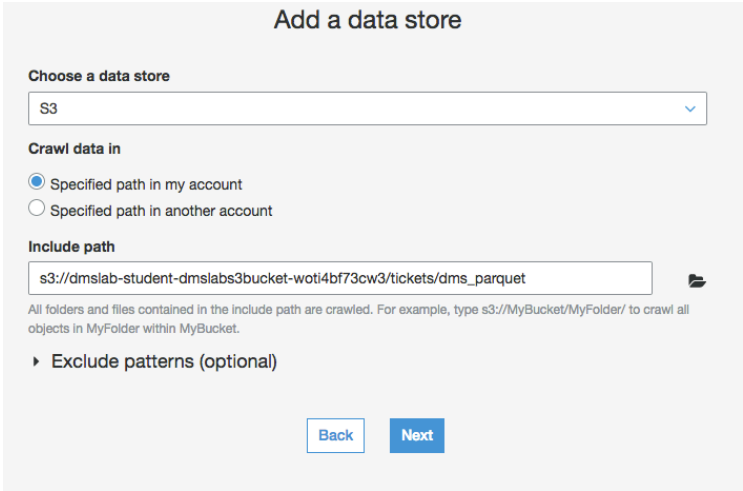
1. In the AWS Glue navigation menu, click **Crawlers**, and then click **Add crawler**.



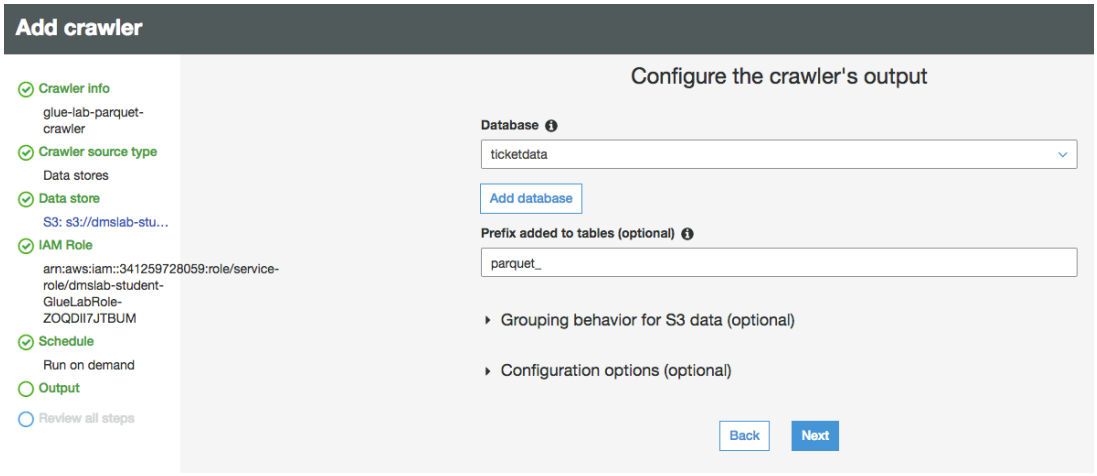
2. For Crawler name, type **glue-lab-parquet-crawler** and Click **Next**.



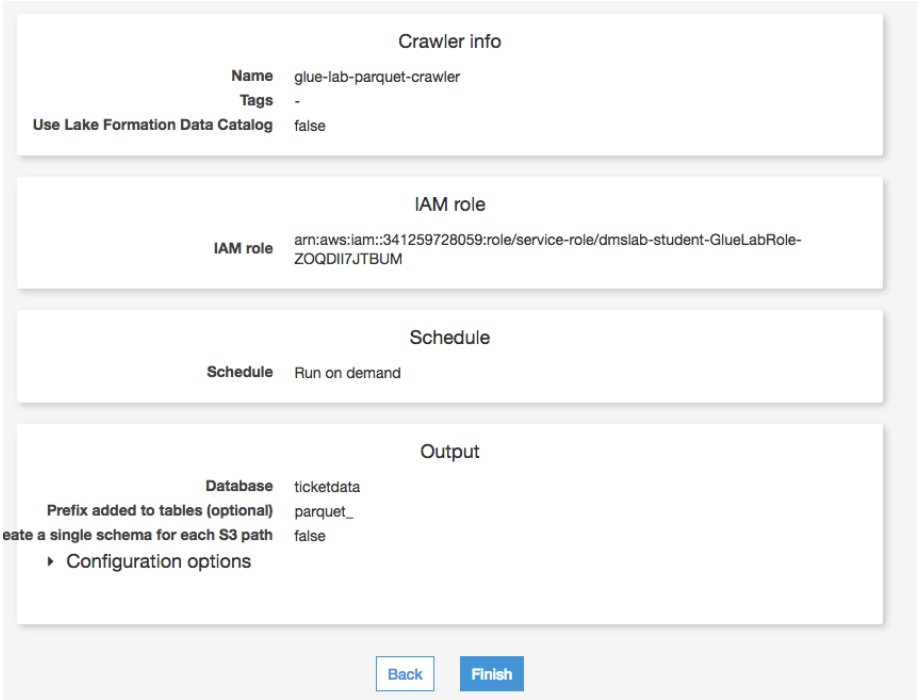
3. In next screen **crawler source type,** select **Data stores** and click **Next.**
4. Choose **S3** as data store type.
5. For Crawl data in, select **Specified path in my account**.
6. For Include path, specify the **S3 root path** that contains the all parquet files e.g., s3://<bucketname>/tickets**/dms_parquet**
7. Click **Next**.



8. For **Add another data store**, select **No** and Click **Next**
9. On the Choose an IAM role page, select **Choose an existing IAM role**.
10. For **IAM role**, select the role "xxxx-GlueLabRole-xxxx" from drop down and Click **Next**.
11. Select **Run On Demand** in Frequency and Click **Next**.
12. Choose the existing **ticktdata** database as the crawler's output database
13. In the **Prefix added to tables (optional),** type **parquet_**

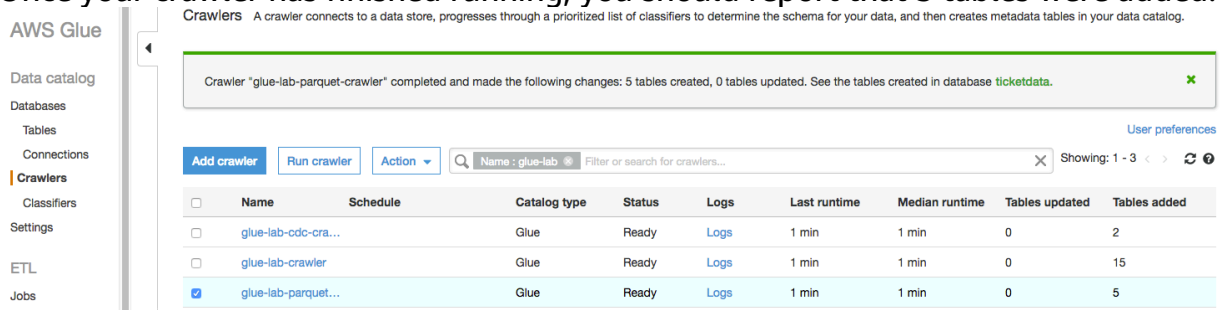


14. Review the summary page and click **Finish**.



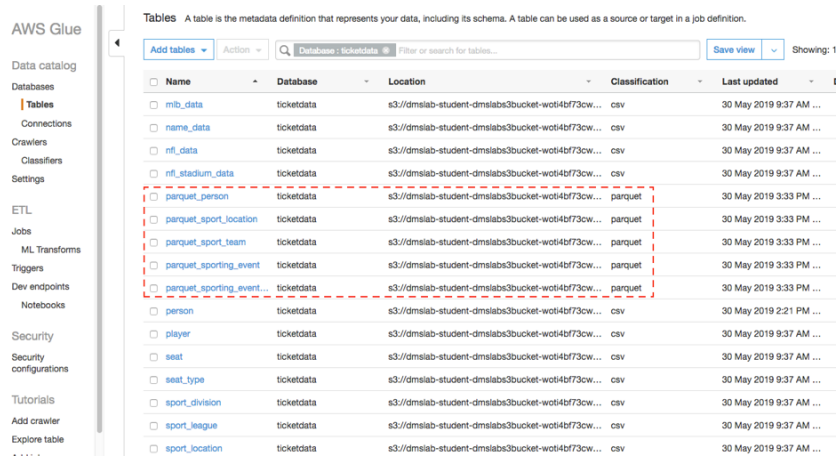
15. On the notification bar, click **Run it now**.

Once your crawler has finished running, you should report that 5 tables were added.



Confirm you can see the tables:

1. In the left navigation pane, click **Tables**.
2. Add the filter "**parquet**" to return the newly created tables.



Next Steps

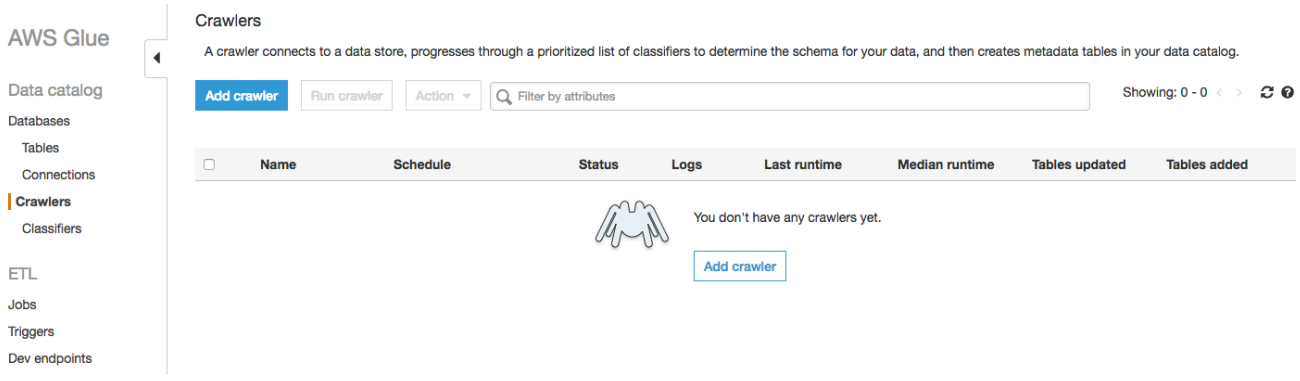
In next lab, we will complete the following tasks:

- Query data and create a View with Athena
- Build a dashboard with QuickSight

Create Crawler for ongoing replication (optional)

Now, let's repeat this process to load the data from change data capture.

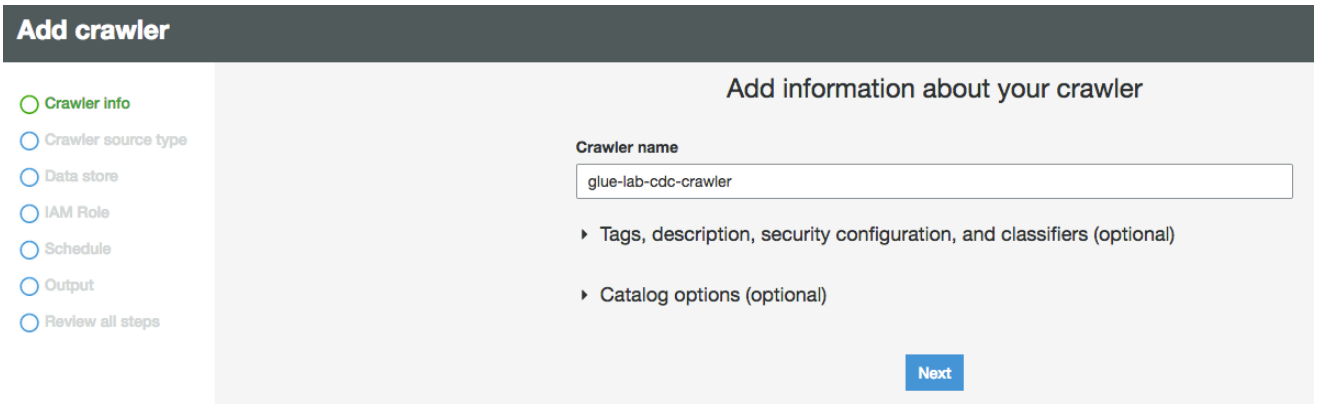
1. On the AWS Glue menu, select Crawlers.



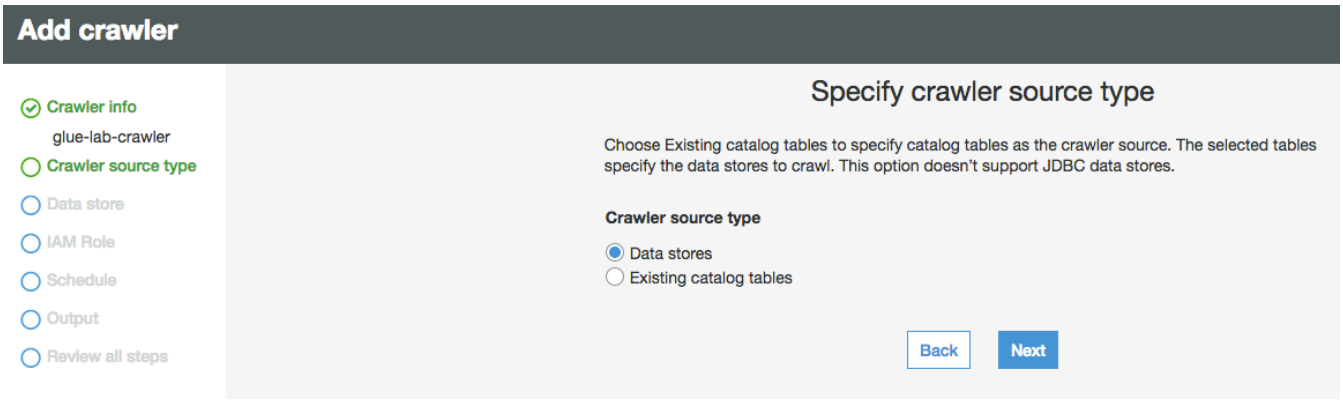
2. Click **Add crawler**.

3. Enter the crawler name for ongoing replication. This name should be descriptive and easily recognized (e.g., " glue-lab-cdc-crawler").

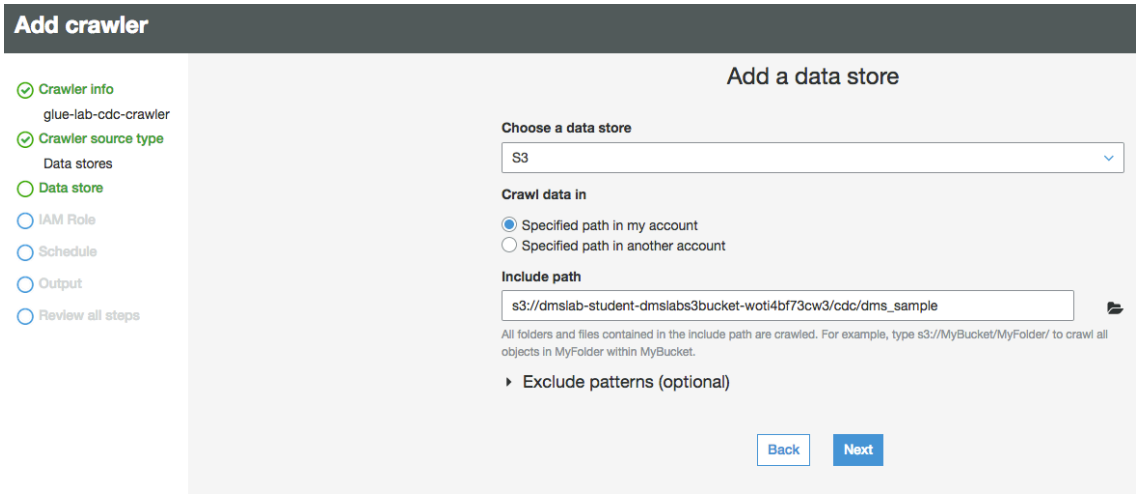
4. Optionally, enter the description. This should also be descriptive and easily recognized and Click **Next**.



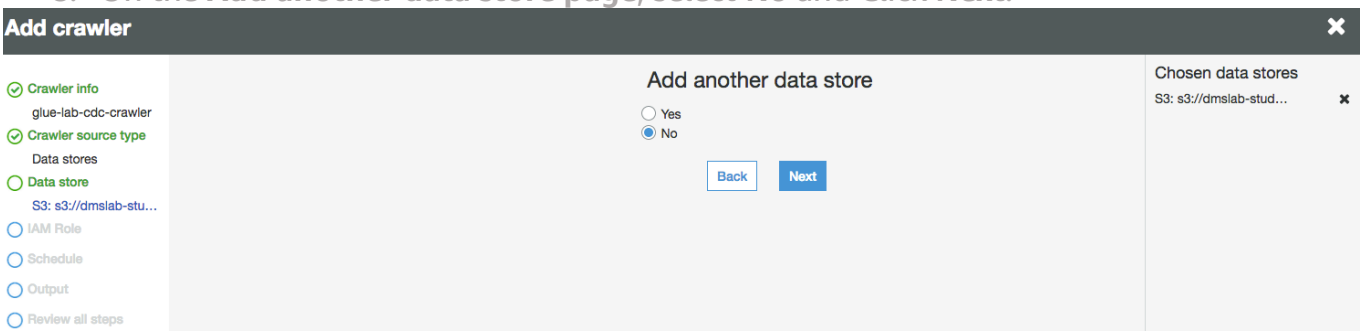
5. Choose **Crawler Source Type** as **Data Source** and Click **Next**



6. On the Add a data store page, make the following selections:
    a. For Choose a data store, click the drop-down box and select S3.
    b. For Crawl data in, select Specified path in my account.
    c. For Include path, enter the target folder for your DMS ongoing replication, e.g., “s3://dmslab-student-dmslabs3bucket-woti4bf73cw3/cdc/dms_sample”
7. Click **Next**.

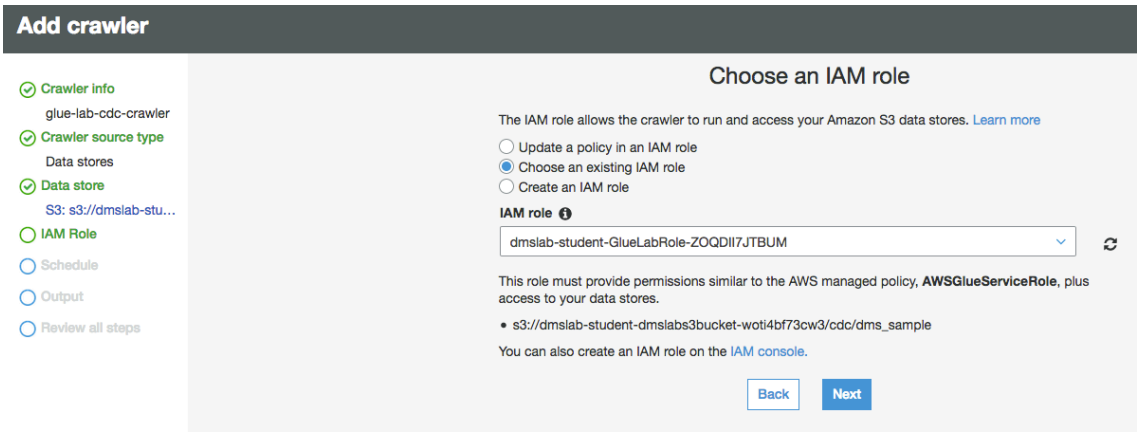


8. On the **Add another data store page**, select **No** and Click **Next**.

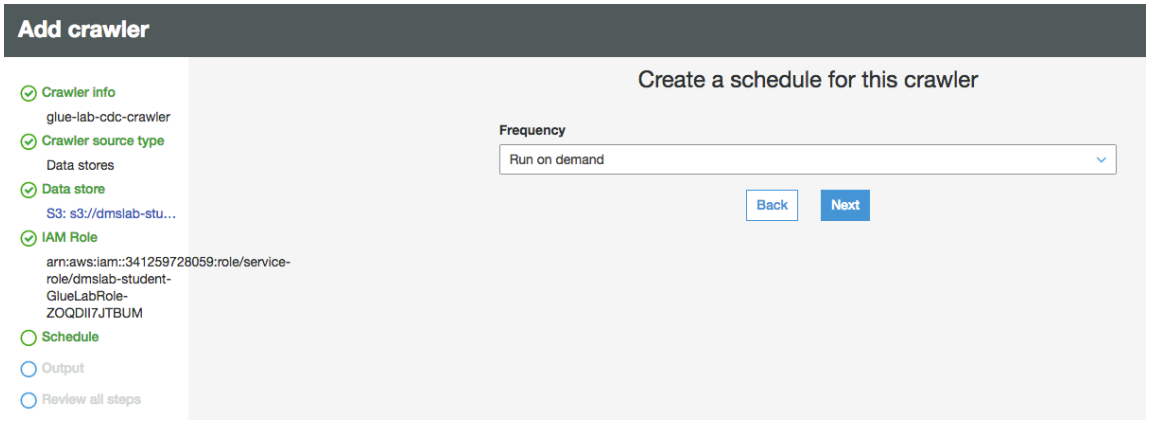


9. On the **Choose an IAM role** page, make the following selections:
   a. Select **Choose an existing IAM role**.
   b. **For IAM role**, select **<stackname>-GlueLabRole-<RandomString>**. E.g. "dmslab-student-GlueLabRole-ZOQDII7JTBUM"

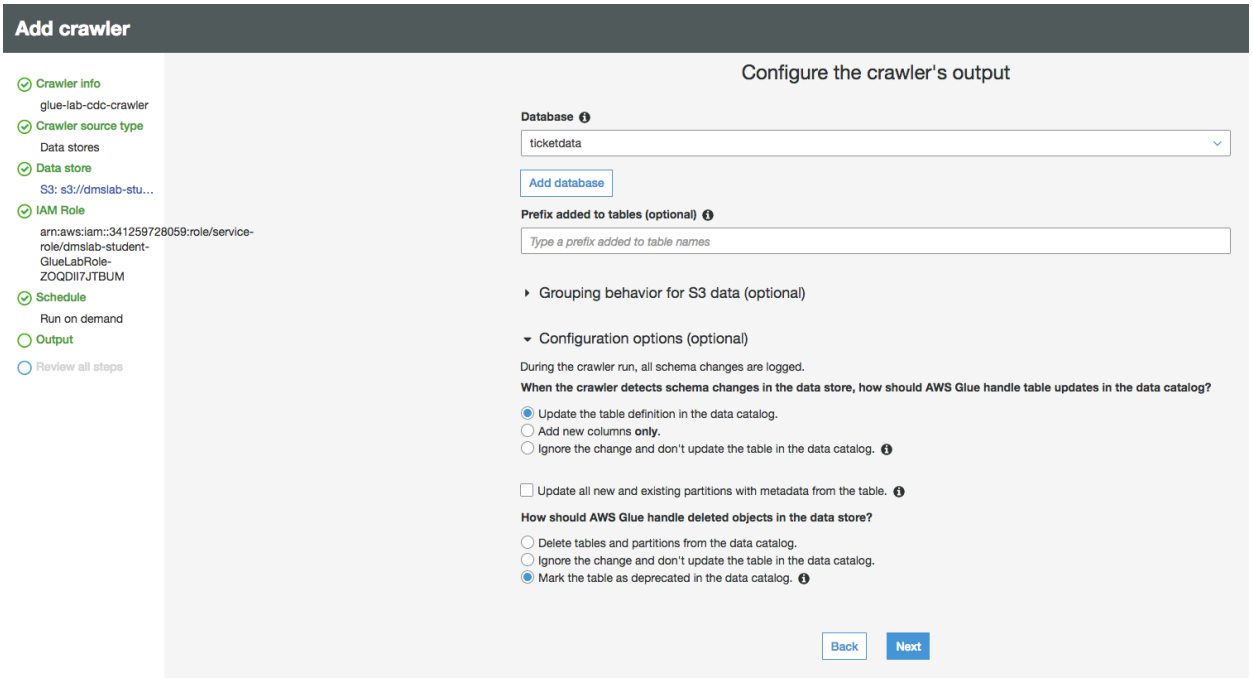
10. Click **Next**.



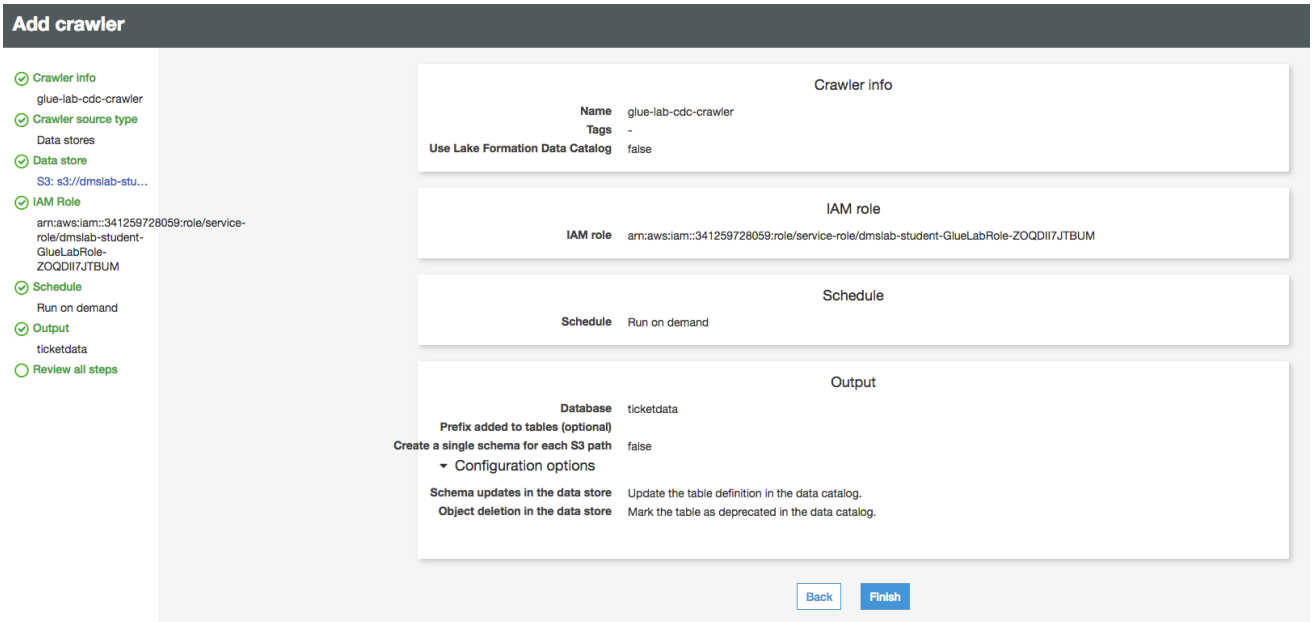
11. On the Create a schedule for this crawler page, for Frequency, select **Run on demand** and Click **Next**.



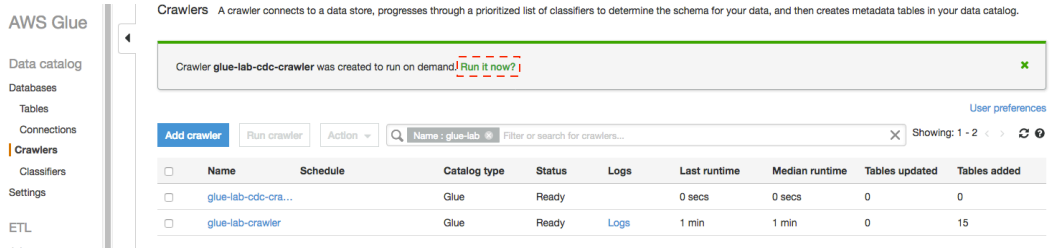
12. On the Configure the crawler's output page, select the existing **Database** for crawler output (e.g., "ticketdata").
13. For **Prefix added to tables (optional)**, specify "cdc_"
14. For Configuration options (optional), keep the default selections and click **Next**.



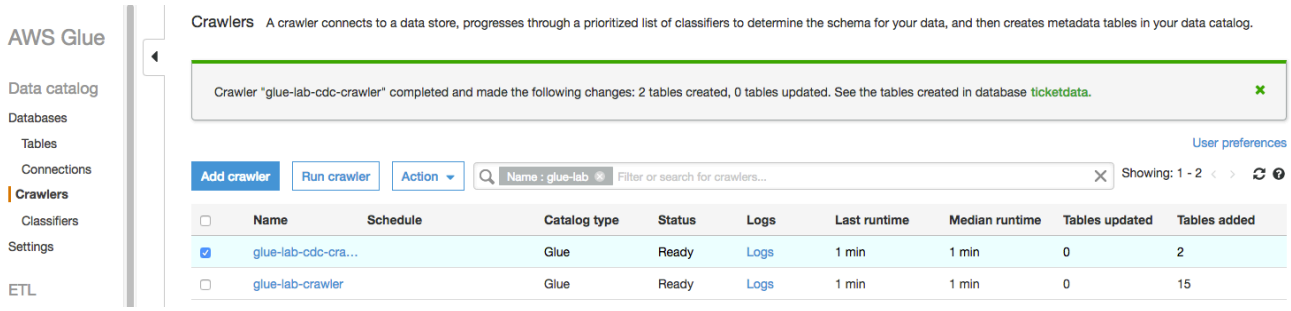
15. Review the summary page noting the Include path and Database target and Click **Finish**. The crawler is now ready to run.



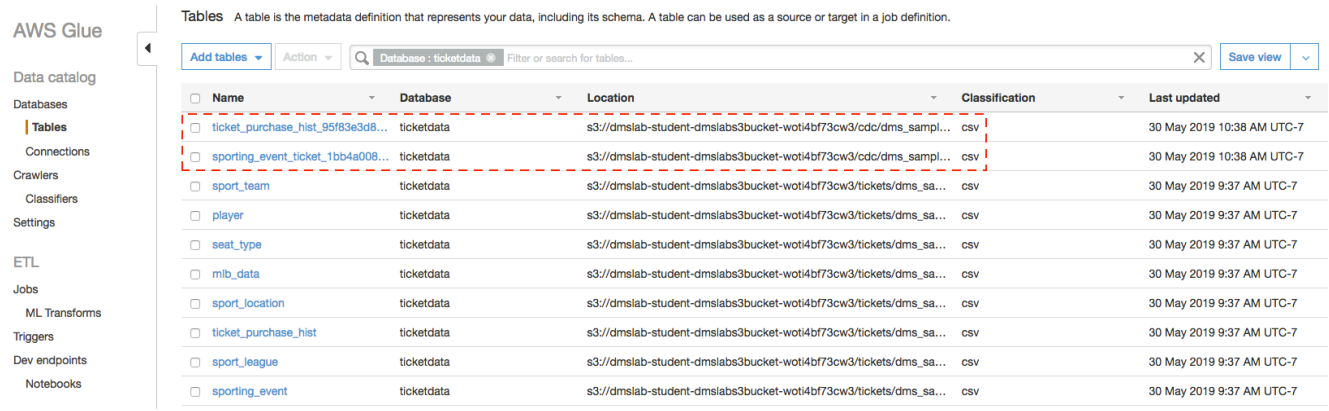
16. Click **Run it now**.



17. When the crawler is completed, you can see it has “Status” as **Ready,** Crawler will change status from starting to stopping, wait until crawler comes back to ready state, you can see that it has created 2 tables.



18. Click the database name (e.g., "ticketdata") to browse the tables. Specify "cdc" as the filter to list only newly imported tables.



You can repeat same steps for CDC data as you preformed for initial full load data which include:

- Create folder structure in S3 bucket to store CDC parquet file.
- Create and Run ETL job to convert csv data into parquets format.
- Create and run another crawler to create data catalog for CDC parquet files.

When you are building an enterprise use cases, it's become important to automate entire pipeline and add notification. Please refer below blogs to try out end to end servlets datalike automation:

Build and automate a serverless data lake using an AWS Glue trigger for the Data Catalog and ETL jobs:

https://aws.amazon.com/blogs/big-data/build-and-automate-a-serverless-data-lake-using-an-aws-glue-trigger-for-the-data-catalog-and-etl-jobs/