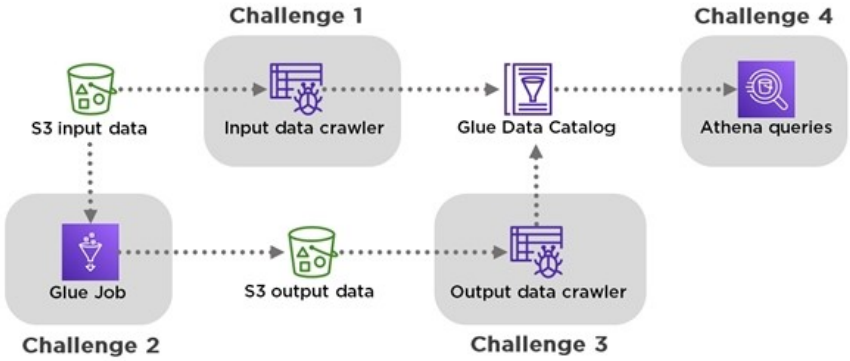
*Transform Data Using AWS Glue and Amazon Athena

Crawl Input Sales Data on S3
You received a CSV file with 100k sales records from the Globomantics Corporation. You need to achieve two objectives. First, transform the file into a Parquet file compressed with Snappy, so that it takes less storage and can be used by other Globomantics departments. Second, analyze the data and return the total sales amount.

This diagram shows the high-level pictures of how the challenges fit together. The S3 input data has 100k lines with sales details. These challenges will help you analyze the input sales data using a high-performance and low-cost approach.



The first challenge is about creating a Glue crawler that crawls the sales input data on S3 and writes details about it into the Glue Data Catalog.

Here is what the input sales data looks like:

id,name,amount

1,Diego,671

2,Cristiano,405

…

100000,Roddy,652

Click the Open AWS console button to the right of these instructions, then use the provided credentials to log in.

Make sure you are in the proper region – the region selector is in the top right hand corner and it should display Oregon; if not, set the region to US West (Oregon).

Use the search box at the top of the page to navigate to the AWS Glue service.

From the left-hand menu underneath Data catalog, click on Crawlers.

On the Crawlers page, click on Add crawler.

Under Crawler name, enter input-data-crawler and click Next.

Leave defaults on the Specify crawler source type page and click Next.

On the Add a data store page, notice that S3 is already selected as a data store and we need to specify the Include path, as follows:

- Click on the folder icon at the right of the textbox that displays s3://bucket/prefix/object. After clicking, a dialog with Choose S3 path appears.

- Click on the + sign next to the aws-glue-athena-lab-... bucket.

- Click on input-data to select it.

- Click on the Select button. This will select the S3 path to the input sales data and close the dialog.

- Back in the Add a data store page, click the Next button.

On the Add another data store page, leave the default No selected and click Next.

On the Choose an IAM role page, notice Create an IAM role is already selected and it needs a name. In the text box displaying Type a name…, type SalesData and click Next.

On the Create a schedule for this crawler page, leave Run on demand selected and click Next.

On the Configure the crawler’s output page, you define a database in the Glue Data Catalog to store the crawler’s output. Click on the Add database button, under the Database name type sales-database, and click Create. The sales-database is now selected in the Database field, so click Next.

In the page with an overview of all steps, scroll down and click on the Finish button.

On the Crawlers page you can see the input-data-crawler and its Ready status. Click on the checkbox next to input-data-crawler,

then click on the Run crawler button above it.

Watch the Status field of input-data-crawler as it changes to Starting. After about a minute, it changes to Stopping. Wait about a minute more until it changes back to Ready. The last column on the right is named Tables added and it indicates that one table is now added to the Glue Data Catalog.

Congratulations for successfully adding a table to the Glue Data Catalog! To look at the new table, on the left-hand AWS Glue column menu, under Databases, click on Tables and you can see the input_data table. Click on the input_data table and scroll down to the Schema section. Notice the three column names (id, name, amount) and their data types. You did not specify them manually, instead the Glue Crawler recognized them automatically for you by crawling the input sales data on S3.

Convert Sales Data with a Glue Job

The original raw input CSV file is about 1.8MB in size. However, the IT policy of Globomantics states that the Parquet format is preferred, to better support complex data processing across all departments, including saving on storage to lower costs. You need a new Glue Job to convert the CSV sales data into the Parquet format.

On the left-hand AWS Glue menu, under the ETL section, click on Jobs.

On the Jobs page, click on the Add job button to start the job creation wizard.

On the Configure the job properties page, under Name, type convert-to-parquet. Under the IAM role label, click on the drop-down and select AWSGlueServiceRole-SalesData, which you created previously. Scroll down and click on the Next button.

On the Choose a data source page, select the input_data table that you created in the previous challenge, and click on the Next button.

On the Choose a transform type page, leave the Change schema option selected and click on the Next button.

On the Choose a data target page, click on the Create tables in your data target option. Under Data store, click on the drop-down and select Amazon S3. Under Format, click on the dropdown and select Parquet.

Under Target path, click on the small folder icon to the right of the text box. In the Choose S3 path dialog, click on the + sign next to

the aws-glue-athena-lab-... bucket, select the output-data folder and click on the Select button. The Target path is now set to the output-data S3 path, so click on the Next button.

On the Output Schema Definition page, you have the option to delete some columns from the target or re-arrange target columns. For now, leave defaults and click on the Save job and edit script button.

Press Esc or click on the small top-right x to dismiss the Script editor tips dialog.

On the script editor page, click on the Save button at the top of the page.

On the left of the page, you can see a diagram with boxes and arrows. From top to bottom, you can see the input_data table from the sales-database database, three transformation steps, and finally the output-data path on S3. The diagram summarizes the code in the editor, which can be used to add custom logic.

Keep the current code and close the editor by clicking on the top-right X label of the script editor, next to the label with a question mark.

You can now see the list of Glue Jobs. Select the convert-to-parquet job, then click on the Action button with a drop-down, and click the Run job action. A dialog with Parameters (optional) appears, click the Run job button in the dialog.

Again, you can see the list of Glue Jobs. Select again the convert-to-parquet job. More details appear at the bottom of the page with details about the job. The History tab shows a table that starts with Run ID. Wait until the Run status column changes from the Running to the Succeeded status.

Congratulations for successfully running the conversion Glue job!

Here is how you can check the output of this Glue job. On the top-left of the page, click on Services to see all services, then click on S3 under the Storage section. Next, click on the aws-glue-athena-lab-... bucket, then click on the output-data folder.

Look at the sizes of the Parquet files: in total they occupy around 600KB. This is about 3 times less storage than the input file, so congratulations again for lowering storage costs!

Crawl Output Sales Data on S3

To analyze the data in the Parquet files, you need to crawl the data and save its schema to the Glue Data Catalog. This is quite

similar to the first challenge, in which you crawled the input CSV file.

Navigate back to the AWS Glue service if you're not already there.

On the left-hand AWS Glue column menu under the Data catalog section, click on the Crawlers item.

Click on the Add crawler button to start the crawler creation wizard.

On the Add information about your crawler page, under Crawler name, enter output-data-crawler, and click on the Next button.

On the Specify crawler source type page, keep defaults and click on the Next button.

On the Add a data store page, under the Include path label, click on the small folder icon at the right of the text box. A Choose S3 path dialog appears. Click on the + sign next to the aws-glue-athena-lab-... bucket, then select the output-data folder, and click on the Select button.

The S3 path to the output-data folder is now entered in the Include path field. Click on the Next button.

On the Add another data store page, click Next.

On the Choose an IAM role page, click to select the Choose an existing IAM role option. Under the IAM role label, click on the Choose an IAM role drop-down, and select the AWSGlueServiceRole-SalesData role. Click on the Next button.

On the Create a schedule for this crawler page, leave the default value and click on the Next button.

On the Configure the crawler’s output page, under the Database label, click on the Choose a database... drop-down, and select sales-database. Click on the Next button.

On the review page, scroll down and click on the Finish button.

The new crawler is created and you are now back to the Crawlers page. Click the checkbox next to the output-data-crawler crawler, then press the above Run crawler button.

Watch the Status field of output-data-crawler as it changes to Starting. Wait until it changes to Stopping, and then back to Ready. The last column on the right is named Tables added and it indicates that one table is now added to the Glue Data Catalog by output-data-crawler.

Congratulations for successfully adding another table to the Glue Data Catalog!

To check the schema of the new table, on the left-hand AWS Glue column menu, under Databases, click on Tables and you can see the input_data and output_data tables (if you don't see output_data, you may need to click the refresh button on the top-right of the page. Click on the output_data table and scroll down to the Schema section. Notice the three column names (id, name, amount) and their data types. Just like in the first challenge, you did not specify them manually, instead the Glue Crawler recognized them automatically for you by crawling the Parquet files.

Analyze Sales Data with Athena
In previous challenges, you crawled the input CSV file and the output Parquet files, which populated the Glue Data Catalog with details about those files. The beauty of the Athena service is that you can easily run SQL queries against files on S3. You do not need the classic approach of importing those files into a database to run SQL queries. Still, Athena needs to know details about the files on S3, which the Glue Data Catalog can offer. Since you already populated the Glue Data Catalog in previous challenges, now you can use Athena to run SQL queries against either the input or the output sales data on S3.

Use the search bar at the top of the AWS Console to navigate to the S3 service.

Copy the name of the aws-glue-athena-lab-... bucket for use later on.

Now navigate to the Athena service using the search bar at the top of the page.

You are now on the Query editor page. On the left side of the page, you can see the sales-database is already selected under the Database field. Also, under the Tables field, you can see the input_data and output_data tables that you created in the previous challenges.

The center of the Query editor page has a text box in which you can write queries. Above the text box, notice the following message: Before you run your first query, you need to set up a query result location in Amazon S3.

Click on the set up a query result location in Amazon S3 text. In the Settings dialog that pops up, in the Query result location field, do the following. Enter s3:// to show it refers to a S3 bucket. Then, paste the bucket name from your clipboard. Finally, enter /athena-results/ to indicate the folder.

The resulting path should look something like this:

s3://aws-glue-athena-lab-bf060el7/athena-results/

Click the Save button.

In the New query 1 tab of the query editor, enter the following query to look at a few records from the original input CSV file:

SELECT * FROM input_data LIMIT 10

Click the Run query button. A few seconds later, you can see ten records with three columns: id, name and amount.

In the New query 1 tab, delete the query so that the editor is empty. Enter the following query: SELECT * FROM output_data LIMIT 10 and click the Run query button. Expect to get other records, and the same three columns. This confirms that the output Parquet files have the same schema as the input CSV file.

Congratulations for querying both the input and output sales data with Athena!

You can analyze the data with even more queries. Here is how to get the total sales amount. In the New query 1 tab of the query editor, delete the query so that the editor is empty, then enter the following query: SELECT SUM(amount) FROM output_data and click the Run query button. Congratulations again for getting the total sales amount (49961659) and finishing this Lab!