DAP Implementation

Step 5: Data Enriching

In my data enrichment implementation of the AWS Data Analytics Platform. For data cataloging, I used AWS Glue to extract my data tables using crawler. This service offered a convenient place for all metadata and schema data that we used throughout the project.

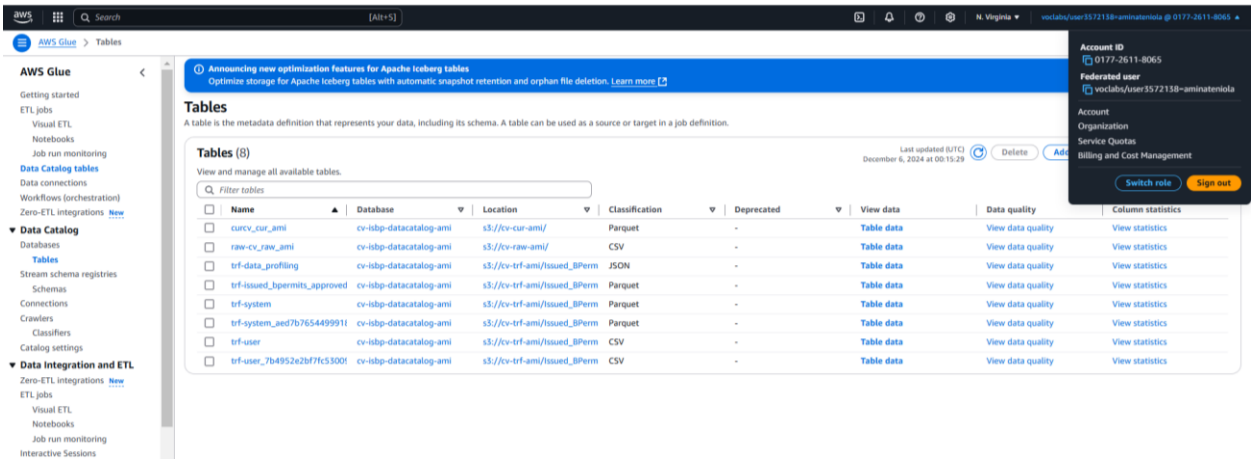

Figure 1: AWS Glue Data Catalog Tables

In the AWS Glue interface shown above, we see different tables that have been created to categorize the permit data. Every table was properly classified and arranged with correct schema definitions and thus the data was easily retrievable for analysis.

Invariably, I started by setting up a strong querying environment in Amazon Athena. The platform had to process permit data, so I created particular SQL queries that would provide valuable information about the types of permits and applicants. To analyze our permit data, I built several analytical queries:

SELECT type_of_work AS permit_category, applicant

```
COUNT(*) AS total_applications, SUM(project_value) AS total_project_value
FROM "trf-system"
GROUP BY type_of_work, applicant
ORDER BY permit_category, total_application DESC;
```

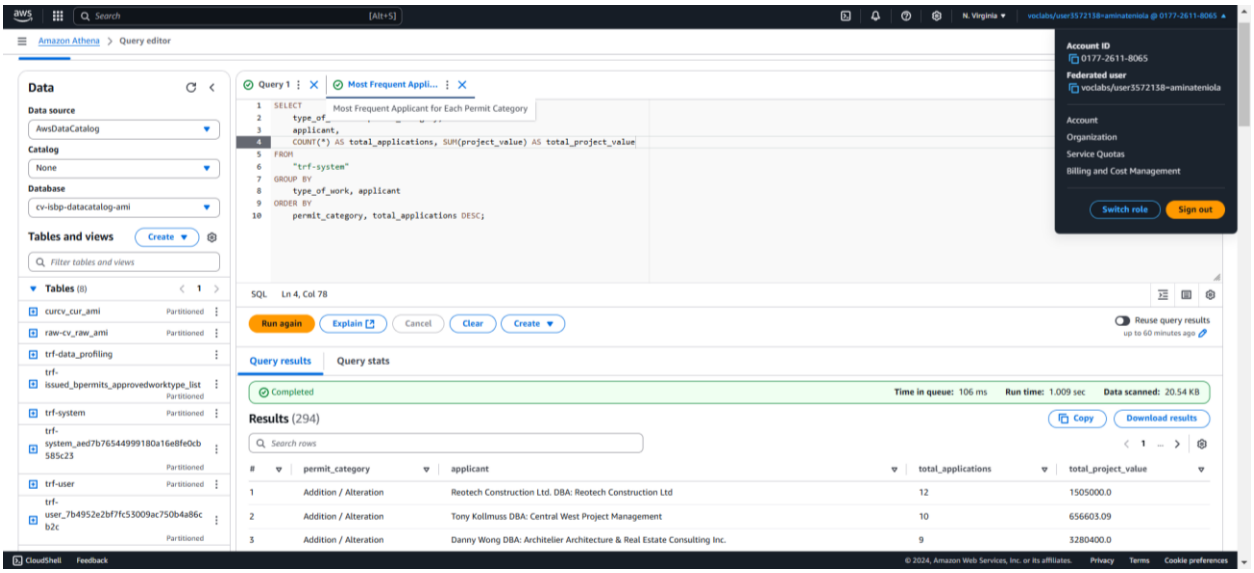

Figure 2: Amazon Athena Query Implementation

The figure above shows the SQL query that I used to investigate the permit categories and applicant information. The query effectively sums up the total values of projects by type of work and applicant, which is helpful in identifying patterns of permit distribution. From this enrichment process, one of the most useful pieces of information was the division of project values by the type of permit. For example, when using the search results, it was found that Addition/Alteration permits had the highest number of applications, and companies such as Reotech Construction Ltd invested a lot of money in different projects.

To organize these queries, the structure within the Athena saved queries was directly saved on my cv-cur-ami bucket so that I could easily retrieve output in my storage.

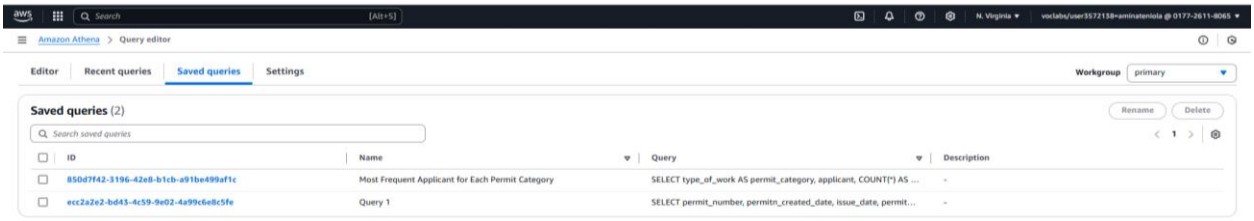

Figure 1: Athena Saved Queries Management

I also had a list of frequently used queries as shown in Figure 2, for instance, the "Most Frequent Applicant for Each Permit Category" which I found helpful in identifying permit distribution. Using AWS Glue's table definitions and Athena's querying functionality, we can understand a robust permit trend and applicant behavior analysis system.

Step 6: Data Protection

In the AWS environment, I began with several of the most important measures for securing our data, which include encryption and access control. The first step in protection, I created a key on AWS KMS.

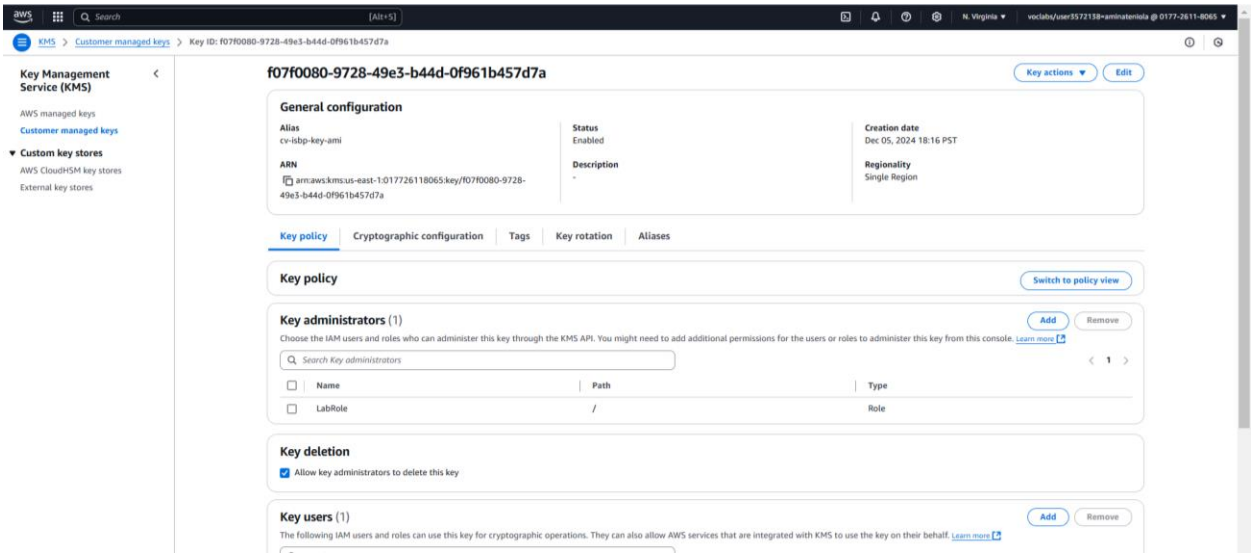

Figure 4: AWS KMS Key

Figure 4 shows the key policy configuration I implemented, the key administrators, and the key users. This good management of the encryption keys will help to see that only those who are allowed can get access to sensitive information. The second step in the data protection plan entailed setting up our S3 bucket encryption options and defining secure data transfer mechanisms.

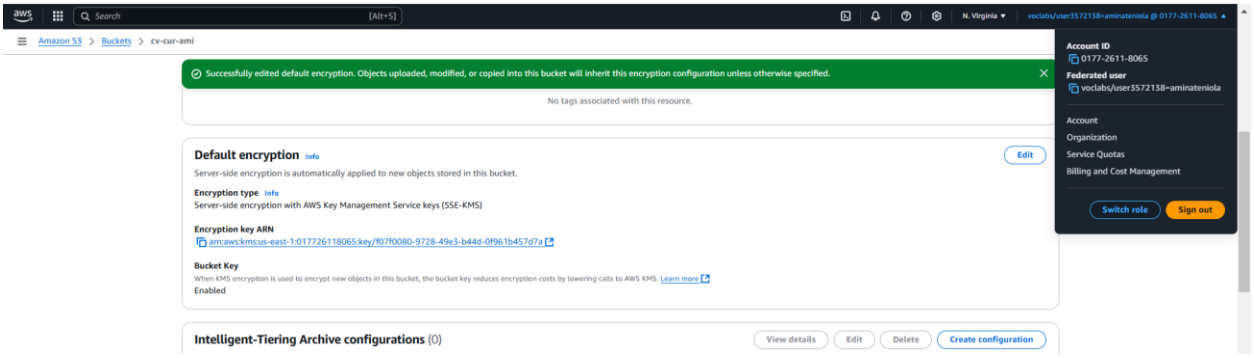

Figure 5: S3 Bucket Encryption Configuration

In the above figure, I set server-side encryption by AWS KMS (Key Management Service) with the created key. This implementation makes it possible for any data stored in our buckets will be automatically encrypted at rest. The encryption settings are the KMS key for data at rest, which is a specific key assigned by the user this key was enabled to the cv-cur-ami bucket in S3 storage.

To improve the approach to data protection, I introduced several levels of protection.

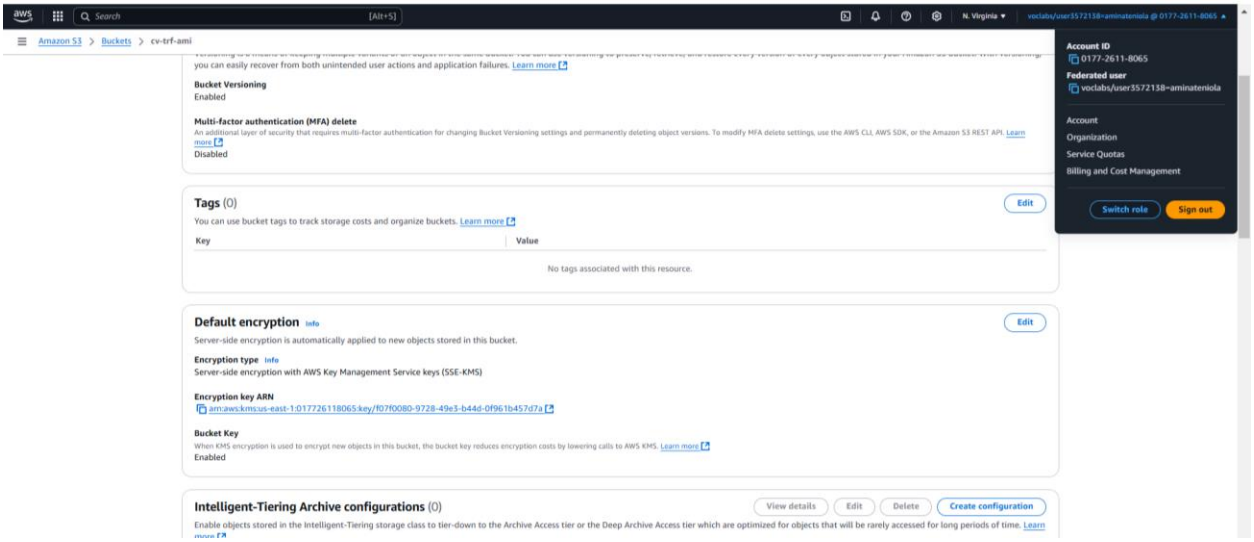

Figure 6: S3 Bucket Security Configuration

In the figure above, I allowed versioning on my bucket to perform a backup and prevent deletions and changes. I also encrypted with the created key on AWS KMS to make sure that all the stored data stays protected.

Step 7: Data Governance

In data governance, I created the ETL (Extract, Transform, Load) plan with AWS Glue Studio. It also helps in enhancing the quality of the data collected and in keeping a check on the standard of data that is used on our platform. The implementation targeted the development of standardized procedures that would ensure data accuracy and at the same time provide

transparency of our data handling procedures.

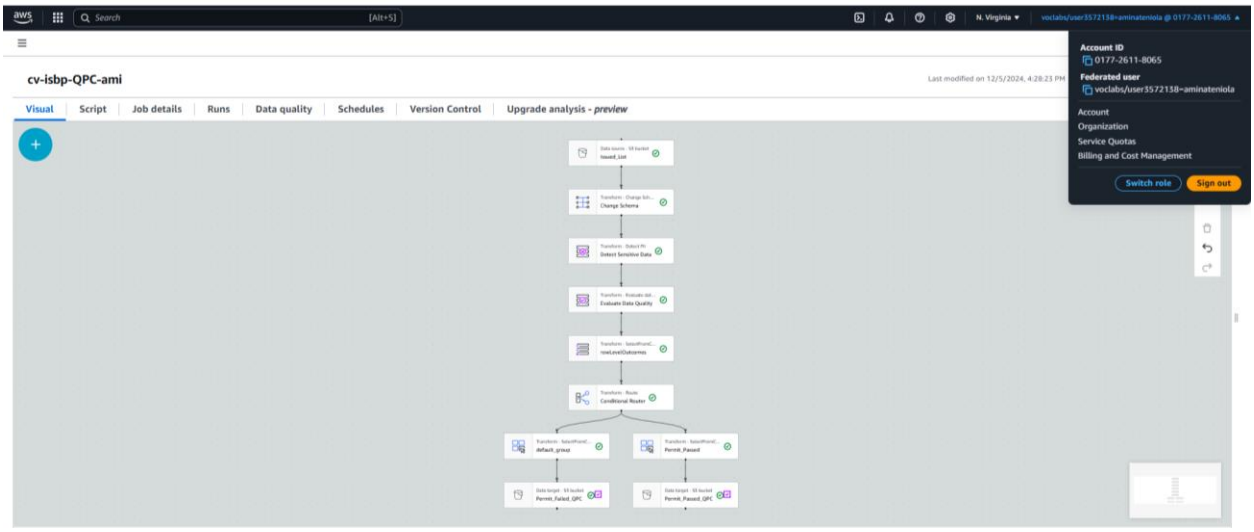

Figure 2: AWS Glue Visual ETL Job

The ETL workflow I designed in AWS Glue Studio, as shown in Figure 7, includes several sequential steps that form a complex web to help us manage our permit data appropriately. The visual workflow starts with a data source node that gathers information from our S3 buckets. Following the extraction, I used a set of transformation nodes that ensure the data is cleaned and normalized for analysis. Each node in the workflow serves a specific purpose - from initial data validation to final quality checks. The visual nature of this ETL job makes it easier to understand and modify the data flow as requirements change.

The transformation process includes several key stages

1. Initial data extraction from source systems
2. Data cleaning and standardization
3. Application of business rules and data validation
4. Quality checks and verification

5. Output generation to destination systems

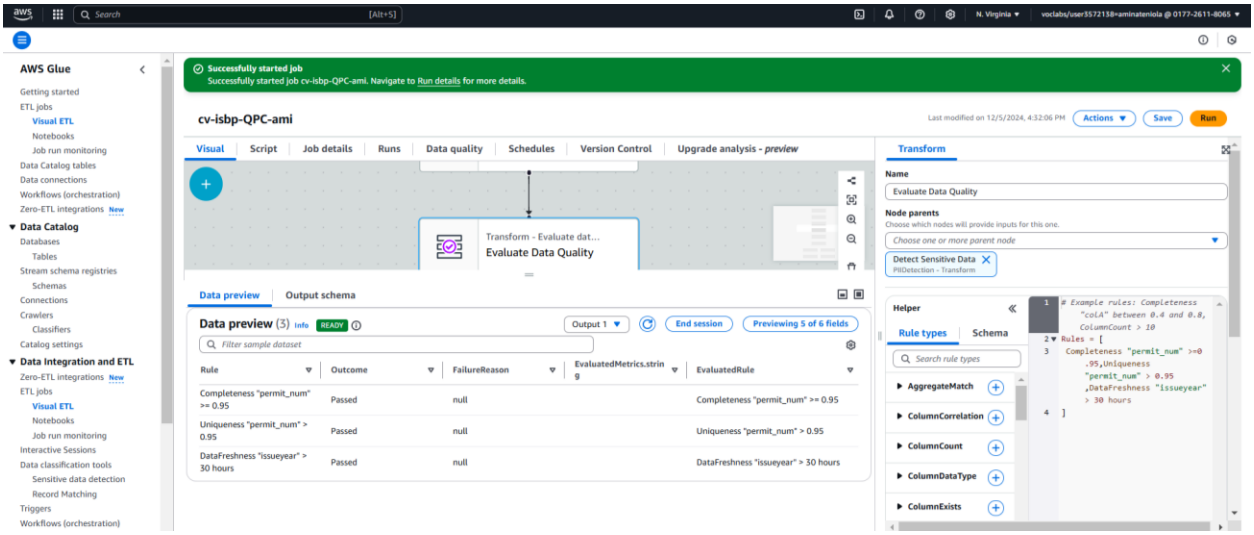

Figure 3: Data Quality Evaluation Results

The data quality evaluation component, visible in the second screenshot, shows the implementation of strict quality and privacy control measures. I established specific data quality rules including:

- Completeness check for permit numbers with a 95% threshold, ensuring that essential permit information is present in our records
- Uniqueness validation for permit IDs with a 95% requirement, preventing duplicate entries that could cause confusion or errors
- Date freshness verification ensuring records are within 30 days, maintaining the timeliness of our permit data

The quality evaluation process automatically flags any records that don't meet these criteria, generating detailed reports on data quality metrics. When records fail to meet the established

standards, they are automatically routed to a separate processing queue for review and correction. This ensures that only high-quality data moves forward in our pipeline.

These governance measures create a robust framework for maintaining data integrity. The visual ETL workflow provides transparency into our data processing steps, while the quality rules enforce our data standards automatically. The implementation also has extensive data transformation logging which allows one to track any problem back to where it originated. With the help of these governance measures, we have designed a mechanism that enables the processing of data. Effectively but at the same time can ensure that the data it processes is of high quality. The automated nature of these checks minimizes the possibilities of human interference while guaranteeing that our data is checked by the laid down procedure standards. This approach has enhanced our capacity to achieve and sustain dependable, precise information for the permit management system of the City of Vancouver.

Step 8: Data Observability

To keep track of the data platform I set up extensive logging and monitoring tools in place. After that in detail allow us to observe the usage of the data on a certain number of days, the performance of a system, and other security problems.

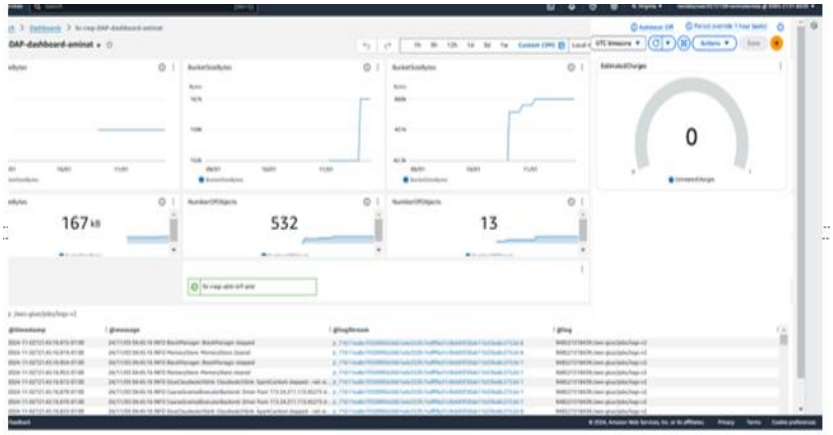

Figure 4: AWS CloudWatch Logs

The logs depicted in the figure above provide information about the different resources used as well as data access patterns. This monitoring assists us in addressing any arising problem effectively as it shows us where they are most rife. These implementations are data protection, governance, and observability, and when applied together, develop a strong and highly secure framework for data analytics.

By monitoring and analyzing the data, we can be sure that the City of Vancouver's data is up to date. Remains protected while at the same time being available for the intended users to make informed decisions on. The implemented systems enable rapid detection of possible problems and have the necessary degree of openness. To be able to change requirements as needed. The whole process adheres to AWS standards and ensures data Data security and management are also in compliance with the AWS standard. Protection regulations. Using CloudWatch for continuous monitoring helps us avoid any issues arising and become a problem. Respond to any issues reflecting both personal and platform effectiveness and security issues.