**Team 2**

**AWS Data Analytics Platform for the City of Vancouver**

Sree Charan (2316748)

University Canada West

Phase 2

**Dataset** – Employee remuneration and expenses

# **DAP Design and Implementation (Employee remuneration and expenses - Sree Charan Addala)**

## **Descriptive Analysis**

City of Vancouver publishes thorough reports on employee pay and costs to show everyone how public servants are compensated. Because the data is from many years and includes extensive financial information, AWS cloud computing allows for flexible and economical storage, processing and analysis. I studied data about Civil Engineer I employees to learn how their pay and work expenses have gone up or down over time. With Amazon S3, Glue, and Athena, I was able to view and analyze these trends using useful metrics. This made it possible for me to notice annual changes, check if the budgeting stayed steady and analyze the alterations in public spending over time and role changes.

A screenshot of a computer

AI-generated content may be incorrect.

## **Step 6: Data Security**

The City of Vancouver dataset Employee Remuneration and Expenses holds sensitive financial information to which extreme care must be taken to ensure data integrity and compliance. According to the privacy laws in Canada, such as PIPEDA and FIPPA, data protection is indispensable and mandatory to prevent data access by malicious actors and malicious websites. To overcome this, one can apply more layers of security to the dataset via the AWS services to protect it during the analytics process.

One of them is to implement AWS Key Management Service (KMS), which will offer encryption control and access management. A special key called employee-remuneration-key was created for this project to encrypt the information that was stored in S3. To make this key, we went to the AWS Management Console, clicked on KMS, and then clicked Create a key. We specified the necessary type as symmetric and gave it to the application for encryption and decryption. IAM roles were created to restrict necessary permissions and administrative access. After reviewing the policy and developing one, the key was set to be deployed. This KMS key means we can encrypt and decrypt confidential financial data on the AWS services, thus achieving greater control and safety.

**Figure 1**

*Key for all the buckets in KMS*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-work

**Figure 2**

*Key for Raw Bucket in KMS*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 3**

*Key for Clean Bucket in KMS*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-work

**Figure 4**

*Key for Curated Bucket in KMS*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

AWS default is the capability to encrypt by using its self-managed keys. However, we have migrated to a dedicated S3 bucket encryption because, after installing our custom key of KMS, our S3 buckets got updated to default encryption. This will guarantee that any data stored in the raw and transformed buckets is encrypted with our dedicated key, providing better supervision and compliance. Bucket versioning was also included to ensure that everything in our S3 buckets has a historical version. This attribute is vital in preventing data loss by mistake, deletion, or overwriting. All changes are kept as independent versions; thus, prior states can be retrieved, compliance needs are met, and business continuity is ensured.

**Figure 5**

*Updated the Key in the Raw bucket once the Key is generated in KMS*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 6**

*Updated the Key in the Curated bucket once the Key is generated in KMS*

A screenshot of a computer

AI-generated content may be incorrect.

*Note: Own-Work*

**Figure 7**

*Updated the Key in the Clean bucket once the Key is generated in KMS*

A computer screen with a message

AI-generated content may be incorrect.

*Note:* Own-Work

To maximize the robustness and availability of our data for all the buckets used in the pipeline, that is, the raw, curated, and transformed buckets, AWS S3 Replication was configured. Such architecture provides high availability and failover in case of disaster due to replicating the data in a different region. The replication procedure was rather typical: (i) turning on the replication rules, (ii) choosing destination buckets, and (iii) checking whether the replication had happened successfully. As we can see in the screenshots, replication has been successfully deployed on all S3 buckets in our Data Analytics Platform (DAP).

**Figure 8**

*Updated Replication Rule in Raw Bucket once the key is enabled*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-work

**Figure 9**

*Updated Replication Rule in Curated Bucket once the key is enabled*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 10**

*Updated Replication Rule in Clean Bucket once the key is enabled*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

## **Step 7: Data Governance**

Data governance entails ensuring the data's accuracy, organization, and security. Good governance activities enable the City of Vancouver to make sound judgments, prevent processing errors, and preserve the privacy and integrity of the government's financial records. In the case of the Employee Remuneration and Expenses dataset, proper data governance will make sensitive compensation data structured and managed, with clear management and supervision, which will allow the company to operate well and report credibly.

As depicted in the figure below, one of the Visual ETL jobs built based on AWS Glue relates to processing the remuneration and expenditure records. My role in this position was to pay attention to the freshness of data and its uniqueness, which is more critical for trustworthy reporting and comparability between the years. Such checks ensure that the data is updated and contains no duplicate copies or null values, which is necessary when going through financial data.

My method of creating the ETL job started with landing on AWS Glue and creating a visual ETL through the AWS Management Console. After getting to the workspace, I renamed the job, saved it under a suitable name, and corrected all access control errors by allocating the proper IAM lab role. Then I chose the data as Amazon S3 and indicated the raw dataset bucket where the .csv files are located. I applied governance rules by transforming the following components: Evaluate Data Quality and Conditional Router. I have new data quality requirements to check up on completeness (no null fields) and Freshness (last available year). Its consideration upon these appraisals led to the automatic directing of the records between two branches: one of the successful quality check records and one of the unsuccessful quality check records. The outputs were pushed into two different folders in the S3 transform storage bucket, i.e., /passed and /failed. Such a division will make monitoring and re-processing rejected records easier, and only legitimate and authoritative data will flow to the last analytics layer.

**Figure 11**

*Visual ETL in AWS Glue for Data Quality Check*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 12**

*Once Visual ETL is generated, the Job was initiated and it was successful.*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 13**

*CSV is being generated for Successful records*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 14**

*CSV is being generated for Failed records*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

## **Step 8: Monitoring of data**

Data monitoring enables the City of Vancouver to have the rightful records of the employee remuneration and expenses, as they are accurate, secure, and reliable. The city can swiftly detect anomalies, performance-related problems, or the risk of various security threats with real-time alerts and visual tools such as dashboards. Monitoring also guarantees transparency and effectiveness in the management of financial information systems.

I designed a personalized CloudWatch Dashboard using the AWS Management Console to apply the monitoring. I logged into CloudWatch and initiated creating a dashboard by clicking on the link to create a dashboard, where I gave the dashboard's name and added widgets such as line graphs and number indicators. I set up the dashboard to monitor important AWS Glue job runs and S3 storage usage concerning the Employee Remuneration and Expenses dataset. I set the filter to 3 months of job activity and data access frequency. I also configured the S3 buckets to limit the usage level and store the dashboard so that the graphics monitoring configuration is in place. I can see the near real-time dashboard providing a flexible view of the operational status of my data pipeline.

As a form of proactive control, I set up one of the S3 buckets that contains transformed information with a CloudWatch Alarm. To test the alert feature, I set a low threshold value. As had been anticipated, the state of the alarm changed to in alarm, which showed that it is functional and will send alarms in case of overuse, unusual access, or sudden increase in storage. These alarms boost system reliability and expediency.

Finally, I enabled AWS CloudTrail to allow for the traceability of user actions and audits. I logged into the CloudTrail console and clicked the option “Trails,” then I clicked on the option “Create Trail.” I referred to the trail as the employee-remuneration-trail, picked an existing S3 bucket, and the encryption key was my own KMS, which I had previously created. I empowered the pathway throughout various regions and incorporated management activities. Having gone through the settings, I made the trail. This guarantees that all access and actions on AWS with the remuneration dataset will be logged safely. Still, it also helps meet most of the requirements for adherence and internal scrutiny.

**Figure 15**

*Cloud Watch Dashboard is generated.*

A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

**Figure 16**

*Trial created using AWS Cloud Trial* A screenshot of a computer

AI-generated content may be incorrect.

*Note:* Own-Work

Figure 17

*Log of User Activity can be observed in AWS Cloud Trail*

A screenshot of a computer

AI-generated content may be incorrect.

# **DAP Evaluation**

Data Analytics Platform (DAP), under development in the City of Vancouver, is a multifunctional cloud platform that implements Amazon Web Services (AWS) to enable data-intensive administration. The architecture developed together with Team 2 utilizes a variety of datasets related to the city, such as employee remuneration, pavement conditions, business licenses, and council voting records, to improve transparency, efficiency of operations, and decision-making based on data. The platform is designed and implemented using the five core AWS Well-Architected Framework pillars: Operational Excellence, Security, Reliability, Performance Efficiency, and Cost Optimization.

## **Operational Excellence**

This project is based on operational excellence. To get a consistent pipeline across all datasets, the team adopted the Amazon S3 buckets storing raw, cleaned, and curated data, AWS Glue as an initiator of ETL (Extract, Transform, Load) processing, and Athena as a query initiator. Quality checks, including the determination of freshness and completeness of the data, were embedded into Glue jobs to clean each dataset. The checks were automated with the help of the Visual ETL made by Glue, in which records are automatically moved to either the past or failed branches depending on the quality measures. Because of this approach, only credible data was transferred to the analytical layer. Moreover, the Amazon CloudWatch services were able to run real-time dashboards to monitor the job executions of Glue, as well as S3 bucket performance, with alarms triggered to signal out inconsistencies. Such proactive monitoring ensured good visibility and rapid resolution of problems that form the basis of sustainable and scalable operations.

## **Security**

Security was deeply embedded in every stage of the platform. Given the sensitive nature of datasets—such as financial information in remuneration reports and personally identifiable data in business licenses—the team implemented multiple layers of encryption and access control. AWS Key Management Service (KMS) was used to create custom keys for each project module (e.g., employee-remuneration-key, cpi-key-sid). These keys enabled server-side encryption (SSE-KMS) for S3 storage, providing granular control over encryption and decryption processes. IAM roles and policies ensured only authorized personnel could access specific data assets. Moreover, AWS CloudTrail was activated across all modules to log API activities and user access. These logs were also encrypted using KMS and stored securely in S3 buckets. This rigorous approach not only met Canadian privacy regulations such as PIPEDA and FIPPA but also aligned with best practices for industry-grade cloud security.

## **Reliability**

To reinforce the system’s reliability, the team adopted key AWS strategies that ensured fault tolerance and disaster recovery. Bucket versioning was enabled to preserve historical versions of datasets, which allows for data recovery in case of accidental deletion or overwrite. Cross-region replication was set up for all critical S3 buckets, including those for raw, cleaned, and curated data. This replication strategy guarantees high availability by duplicating data across geographically dispersed regions. Alarms in CloudWatch were used to detect changes in storage usage and potential ETL job failures, ensuring the operational health of the platform. Additionally, by separating passed and failed data during quality checks, the team avoided the risk of corrupt data contaminating analytical results. This design promotes both data integrity and continuous availability, critical factors for municipal decision-making platforms.

## **Performance Efficiency**

The efficiency of the performance was possible with serverless AWS and auto-scaling. It relied on AWS Glue and Athena to process and analyze too large datasets without manual infrastructure provisioning. As an example, one can query Parquet-formatted data stored in S3 and use Athena to spend less on the scan and achieve better query performance. Glue jobs were set to run using appropriate Data Processing Units (DPUs) that strike a balance between the speed of running the job and the cost of running the job. Other optimization steps that were carried out in the Pavement Condition Index (PCI) project include the data partitioning according to geographic characteristics and PCI scores that enhanced query execution performance, as well as facilitating a more efficient downstream processing. Lightweight transformation work included a small EC2 machine and a gp3 EBS volume, permitting a small compute environment with minimal cost. The dashboards in CloudWatch provided real-time information about the use of resources, enabling the team to identify and remove inefficiencies simultaneously.

## **Cost Optimization**

The architecture had the optimization of costs as a constant priority. AWS Pricing Calculator helped at the planning stage, providing an estimation of the costs involved with Glue jobs, Athena queries, and S3 storage costs. One of the tactics involved the application of S3 lifecycle policies, whereby data that was not too frequently accessed was transferred to Glacier Flexible Retrieval, making the long-term costs of data far lower. Moreover, datasets have been summarized with ETL jobs to minimize the cost of processing data in querying functions, therefore cutting down the cost of querying data on Athena. The team also set up billing alarms in CloudWatch, including the alarm that went off when expenditure reached over $10 after six hours, to avoid the unexpected costs. Those aggressive cost-containment efforts played a critical role in achieving the balance between functionality and scarce budgets, making the platform able to grow without breaking the budget.

The DAP project organization had a consistent logic of the architecture of each module, and on the other hand, allowed dataset-specific manipulations. Remuneration analysis was carried out wherein transformation rules were used to monitor the fluctuations in salaries as well as work expenses with time. PCI analysis entailed the classification of road conditions depending on their severity and providing statistical overviews to reveal the weaknesses of infrastructure. The business license dataset was narrowed and classified to track the pet service provider business coverage trends, whereas the council voting data was adopted to track the patterns of activities of the councillors. These application cases showed how this architecture was flexible to manage different sets of data and continually achieve the rigour of the architecture in every project area.

# **Conclusion**

In summary, the Cloud computing application in the case of the City of Vancouver is a powerful, secure and scalable DAP platform. The platform meets the requirements of operational excellence by automating the workflow and real-time monitoring as it follows the recommended practices of the AWS Well-Architected Framework. It is done through strict security using KMS encryption and IAM roles. Versioning, replication and clean versus failed data separation ensure reliability. Performance is enhanced through serverless computing, quick search engines and data partitioning. Lastly, the costs are controlled with the use of lifecycle rules, use tracking, and architectural optimization. The system is one of the role models for other municipalities, with the goal of migrating towards cloud-based analytics to improve its public services.