Graph Database Modelling using AWS Neptune and Gremlin

Business Overview

Instead of utilizing rows and columns in tables, graph data modeling allows users to define an arbitrary domain as a linked graph of nodes and relationships with properties and labels. Graph representation of data has several advantages over relational databases. A graph database may be searched by edge type or by traversing the complete graph. Because the associations between nodes are not computed at query time but are stored in the database, traversing the joins or relationships in graph databases is relatively fast. Graph databases are useful for use cases like social networking, recommendation engines, and fraud detection when you need to construct linkages between data and query these associations with minimum processing time.
In this project, the requirement from our customer (simulated), an Airport Operator, is to have an analytics platform using Graph Modelling to answer questions like:

- Which are the best and worst performer airlines for flight delays?
- Which are the most congested airports on the ground (a.k.a. "taxiing")?
- Which are the busiest airport connections?

After data cleaning and modeling, we will load the airlines' data to the Amazon Neptune graph database. We will also look into ways of querying the data efficiently using Apache Gremlin.

Data Pipeline

A data pipeline is a technique for transferring data from one system to another. The data may or may not be updated, and it may be handled in real-time (or streaming) rather than in batches. The data pipeline encompasses everything from harvesting or acquiring data using various methods to storing raw data, cleaning, validating, and transforming data into a query-worthy format, displaying KPIs, and managing the above process.

Dataset Description

The Bureau of Transportation Statistics of the United States Department of Transportation monitors the on-time performance of domestic flights operated by big airlines. The databases include daily airline data such as flight details, origin and destination, carrier company, delay time, and generic delay explanation.

Tech Stack:

➔ Languages-
  - SQL, Python3, Gremlin

➔ Services -
  - AWS S3, AWS Glue, AWS Athena, AWS IAM, Amazon Neptune, AWS Cloud9, AWS EC2, Apache Spark

Amazon S3

Amazon S3 is an object storage service that provides manufacturing scalability, data availability, security, and performance. Users may save and retrieve any quantity of data using Amazon S3 at any time and from any location.

AWS IAM

This is nothing but identity and access management, enabling us to manage access to AWS services and resources securely. One can create and manage AWS users and groups using permissions to allow and deny their access to AWS resources. It is a feature of AWS with no additional charge.

AWS EC2

A virtual server on Amazon's Elastic Compute Cloud (EC2) for executing applications on the Amazon Web Services (AWS) architecture is known as an Amazon EC2 instance. The Amazon Elastic Compute Cloud (EC2) service allows corporate customers to run application applications in a computer environment. Using Amazon EC2 eliminates the need to invest in hardware upfront, allowing users to create and deploy apps quickly. Amazon EC2 allows users to launch as many or as few virtual servers as they want, set security and networking, and manage storage.

AWS Glue

A serverless data integration service makes it easy to discover, prepare, and combine data for analytics, machine learning, and application development. It runs Spark/Python code without managing Infrastructure at a nominal cost. You pay only during the run time of the job. Also, you pay storage costs for Data Catalog objects. Tables may be added to the AWS Glue Data Catalog using a crawler. The majority of AWS Glue users employ this strategy. In a single run, a crawler can crawl numerous data repositories. The crawler adds or modifies one or more tables in your Data Catalog after it's finished.

AWS Athena

Athena is an interactive query service for S3 in which there is no need to load data, and it stays in S3. It is serverless and supports many data formats, e.g., CSV, JSON, ORC, Parquet, AVRO.

Gremlin

The Apache Software Foundation's Apache TinkerPop has created Gremlin, a graph traversal language. Gremlin may be used with both OLTP and OLAP graph databases and processors. Gremlin is a data-flow language that allows users to express sophisticated property graph traversals in a concise manner. A Gremlin traversal is made up of several steps. On the data stream, a step does an atomic action. At each stage, the items in the stream are transformed, removed, or statistics are computed about the stream.

Amazon Neptune

Amazon Neptune is a fully-managed graph database service that makes it simple to create and run applications that interact with large, interconnected datasets. A high-performance graph database engine lies at the core of Neptune, and this engine is designed to handle billions of relationships while querying the graph in milliseconds. The popular graph query languages Apache TinkerPop Gremlin and W3C's SPARQL are supported by Neptune, allowing you to create searches that effectively explore densely linked datasets. The basic unit of Amazon Neptune graph data is a four-position (quad) element (Subject-Predicate-Object-Graph), which is similar to a Resource Description Framework (RDF) quad.

Agenda

- Understand the concepts of Graph database and modeling
- Upload data to S3 using AWS CLI with partitions
- Run Glue Crawler to create Athena table (external) for initial querying
- Run Glue Job to preprocess data-
    - Extract Edges and Vertices files according to Neptune schema
    - Store the data as compressed Parquet format (Columnar) for quick reads
- Perform Graph Querying using Gremlin to gain insights

Key Takeaways

- Understanding the project Overview and Architecture
- Understanding ETL on Big Data
- Introduction to Staging and Data Lake
- Creating IAM Roles and Policies
- Understanding the Dataset
- Setting up AWS CLI
- Introduction to Graph concepts
- Creating a Neptune Cluster
- Understanding Neptune Graph Data Model - SPOG
- Understanding Apache Gremlin
- Exploring Neptune and Jupyter Notebooks
- Data Preprocessing for Neptune
- Creating Apache Spark Glue Dev Endpoint
- Run Spark app in AWS Glue Job
- Review Spark Output Edges and Vertices
- Networking for Cloud9 IDE
- Bulk Load data from S3 to Neptune
- Analytics using Gremlin Query Language