DSCI 551 COURSE PROJECT, SPRING'26

Phase 1: Project Proposal

Group Members

Chanyoung Kim
Yogita Mutyala

USC Viterbi School of Engineering

Chosen Database System and Motivation

Database System: PostgresSQL with Apache AGE Extension
Motivation: In the domain of financial fraud detection, specifically Anti-Money Laundering (AML), data possesses a dual nature. Account balances, customer information and transaction logs require the strict ACID compliance of a Relational DBMS, while fraud patterns (such as circular trading rings) require the deep traversal capabilities of a Graph DBMS. Instead of maintaining two separate systems, we propose investigating PostgreSQL extended with Apache AGE. This setup allows for analyzing how a traditional Relational Engine can be engineered to support Graph workloads. This project aims to determine if the "Graph-on-Relational" approach is a viable alternative to Native Graph Databases for real-time fraud analysis.

Research Question

This project investigates the following research question:
" Can PostgreSQL with Apache AGE efficiently support graph traversal workloads compared to native relational approaches, and what are the storage and execution tradeoffs of this hybrid model? "

Planned Internal Focus Areas and Motivation

1. **Storage Architecture:** Graph to relational Mapping
   Motivation: To understand the storage overhead of graph data in an RDBMS.
   Plan: Analyze how graph elements such as vertices and edges are stored within PostgreSQL heap files through the Apache AGE catalog (ag_catalog). The study will examine:
   - Physical representation of vertices and edges
   - Property storage using JSONB
   - Namespace organization
   - Storage overhead compared to relational tables

2. **Query Execution:** Cypher vs SQL
   Motivation: To measure the performance cost of simulating "Index-Free Adjacency" on a B-Tree based system.
   Plan: We will compare execution plans generated by:
   - Cypher queries through Apache AGE
   - Recursive SQL queries using Common Table Expressions (CTEs)

   In addition, we will analyze the role of B-Tree indexes on graph tables and evaluate how indexing impacts traversal efficiency, query execution plans, and overall performance.

Evaluation Metrics

The system will be evaluated using the following metrics:
1. Query latency
2. Execution cost from query planner

3. Storage overhead
4. Query throughput under simulated workload
5. Index effectiveness

Preliminary Application Idea

Application Name: Hybrid-AML: Relational Fraud Detection System
Description: A financial monitoring dashboard that detects money laundering patterns in real-time.
Core Features:

1. Smurfing Detection: Identify accounts receiving numerous small deposits that sum up to a large amount (Aggregation focus).
2. Circular Trading Detection: Detect closed loops in transaction chains (e.g., A → B → C → A) which indicate artificial volume inflation (Traversal focus).

Data Description

A synthetic dataset will be generated to simulate realistic financial transactions. Synthetic generation allows controlled workload scenarios and repeatable performance experiments. The dataset will include:

- Customer Table: customer_id (PK), name, risk_tier (low/med/high), created_at
- Account Table: account_id (PK), customer_id (FK), account_type (checking/savings/business), open_date, status (active/closed)
- Transaction Table: tx_id (PK), from_account_id, to_account_id, amount, timestamp, channel (wire/ach/cash/crypto etc), merchant_category (optional), tx_type (deposit/transfer/withdrawal), **is_fraud_label (boolean or enum: normal/smurfing/cycle - Target Dataset)**

Scalability Considerations

Although the implementation will run on a single PostgreSQL instance, the project will discuss scalability limitations and potential challenges when applying this approach to large-scale transaction networks.

Risks and Challenges

Potential Challenges include:

- Complexity of Apache AGE configuration
- Difficulty tuning graph queries
- Performance variability with synthetic data
- Limited ability to simulate large-scale production workloads

Team Information and Responsibilities

Team Member	Responsibility
Chanyoung Kim	• Database setup and configuration • Storage architecture analysis • Query execution analysis • Indexing experiments
Yogita Mutyala	• Dataset generation • Performance evaluation • Visualization and reporting

Expected Deliverables

- Working fraud detection prototype
- Performance evaluation results
- Storage and query execution analysis
- Final report documenting findings

Initial References

1. **Apache AGE Documentation.** *The Apache Software Foundation*. https://age.apache.org/
2. **PostgreSQL Global Development Group.** *PostgreSQL 16 Documentation: Chapter 73. Database Physical Storage*.
3. **Stonebraker, M.** (2015). *The Case for Polystores*. IEEE Data Engineering Bulletin. (Context on hybrid data models).