

Optimizing Query Execution In Large Differential Factbase

Student: Foo Chuan Sheng

Supervisor: Asst Prof Li Yi

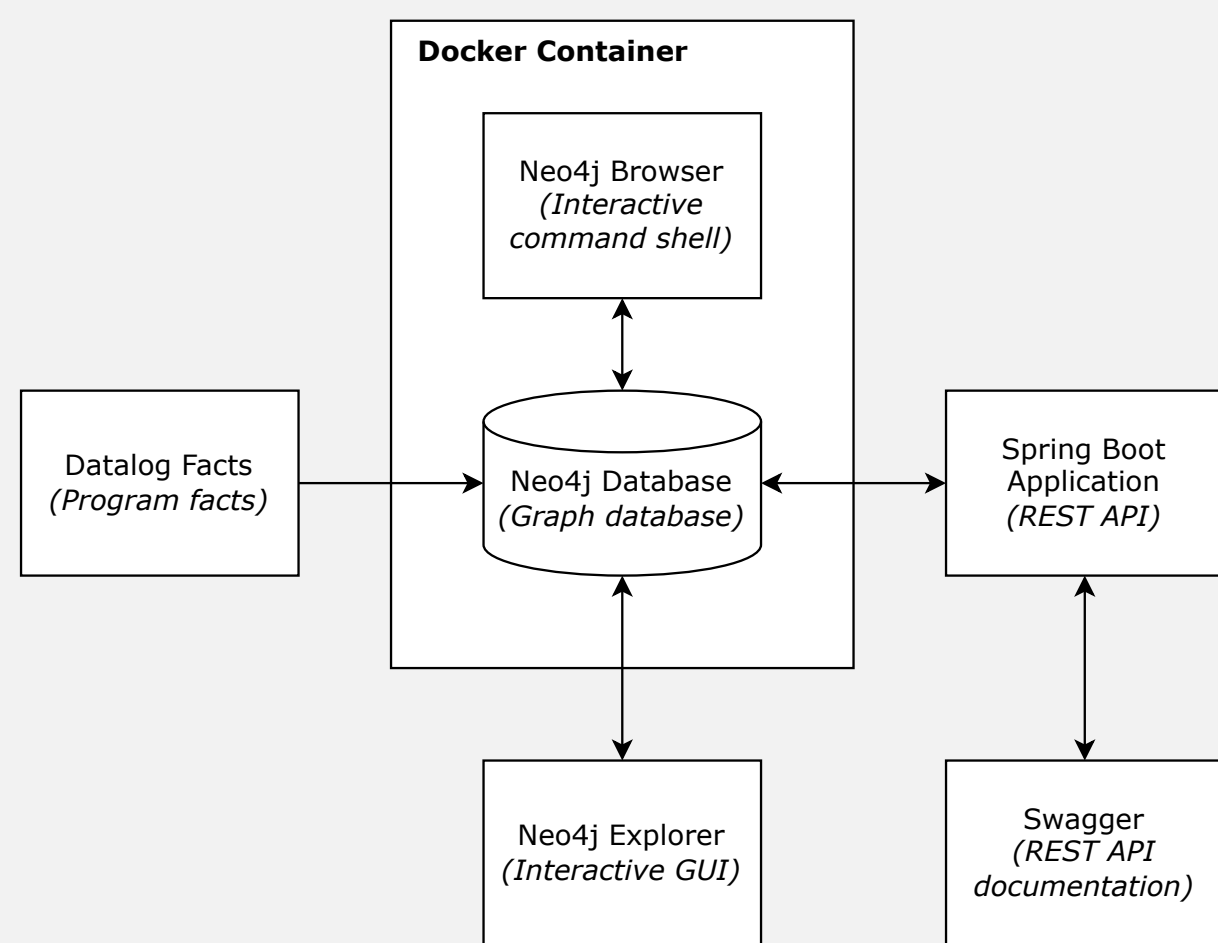
Background

Differential factbase is a uniform exchangeable representation supporting efficient querying and manipulation. It is used to store relevant information of software changes (program facts) such as create and delete operations.

Objectives

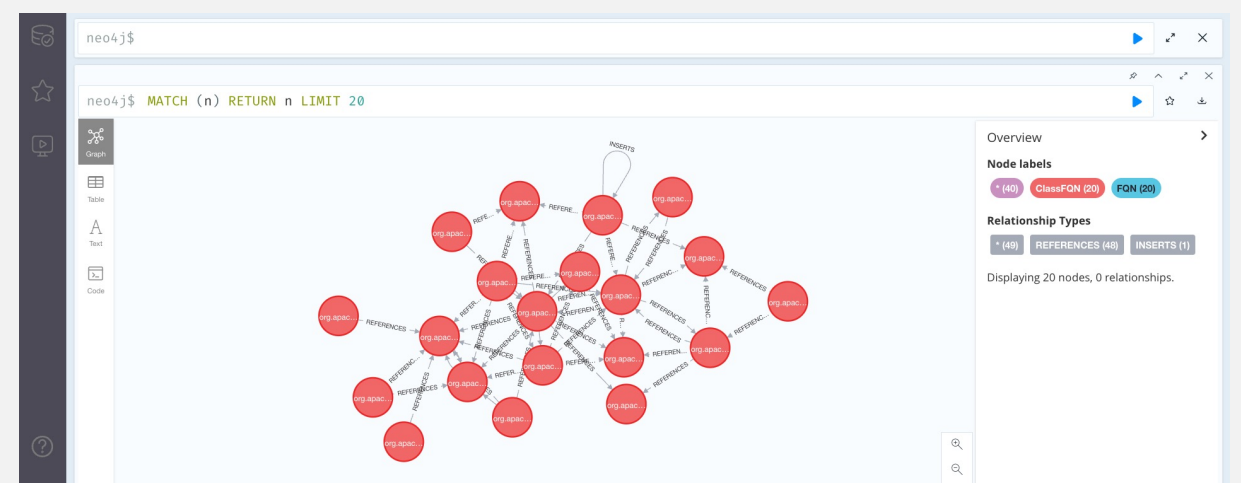
- Create a **system** which **utilizes a graph database** to efficiently store program facts.
- Conduct **benchmark** to check whether **graph database** provides faster query execution times.

System Architecture

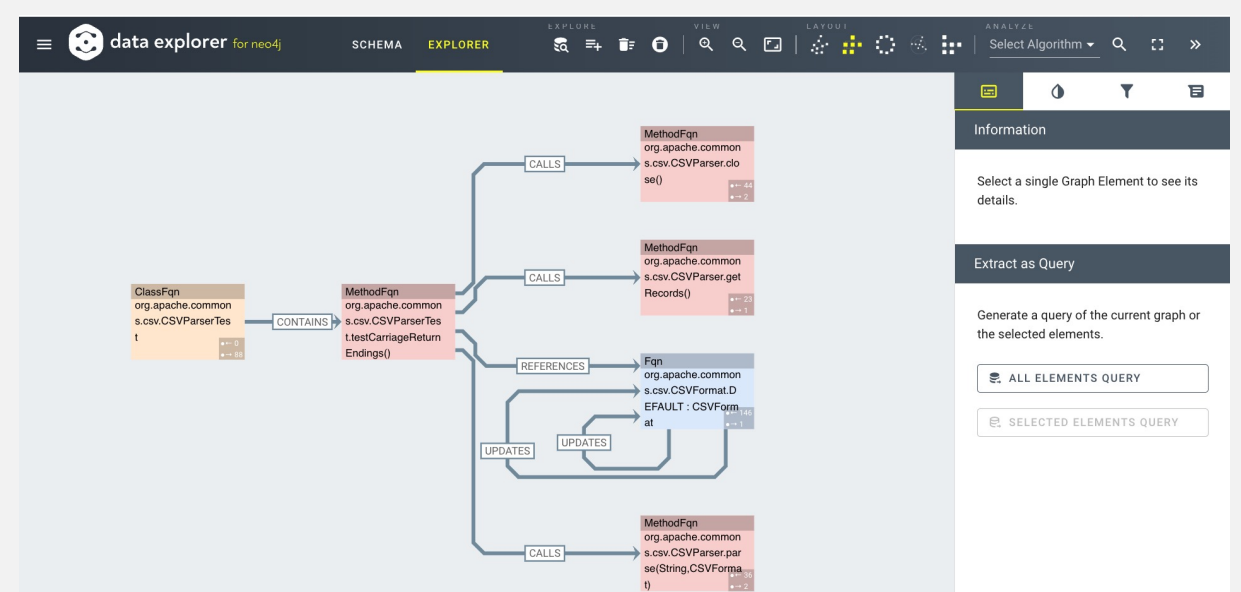


User Interfaces

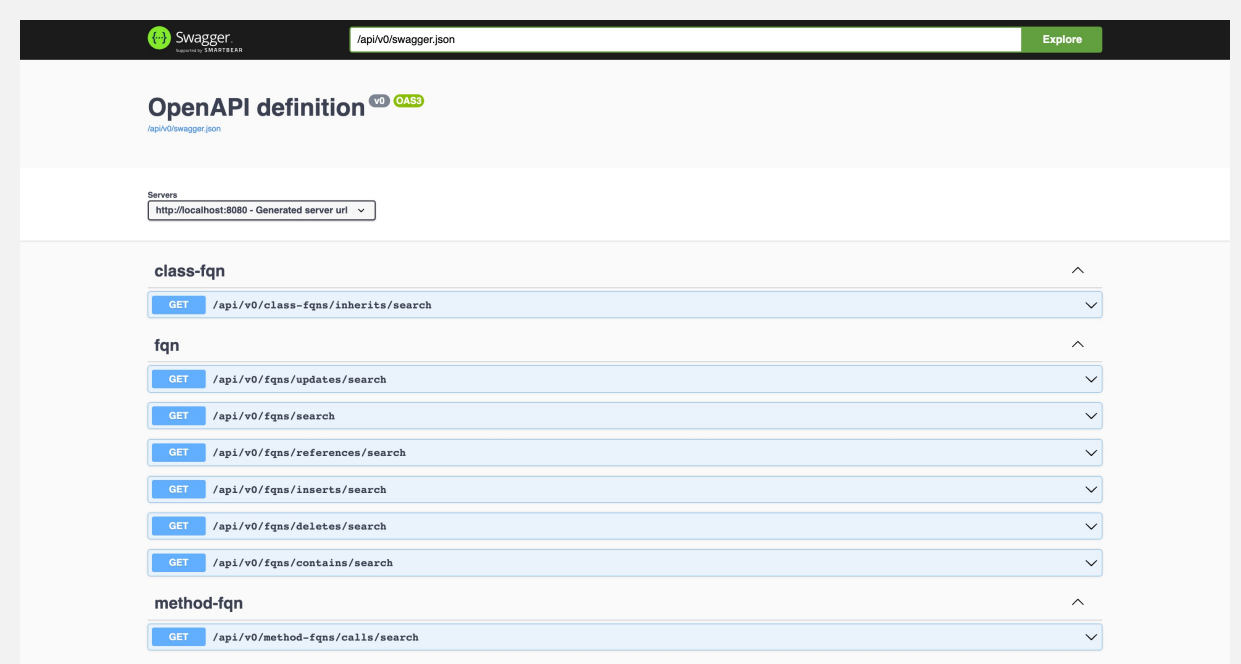
Neo4j Explorer (Interactive command shell)



Neo4j Browser (Interactive GUI)



Swagger (REST API documentation)



Benchmark

Query	Querying Engine	Query Execution Times (ms)				
		1st	2nd	3rd	4th	5th
Find methods with at least 10 callees	Soufflé	60	60	60	60	60
	Neo4j	285	29	20	17	16
Find methods with at least 10 callers	Soufflé	60	60	60	60	60
	Neo4j	295	34	27	16	15
Find method	Soufflé	50	50	50	50	50
	Neo4j	57	2	2	3	2
Find all callees of a method	Soufflé	60	60	60	60	60
	Neo4j	114	4	4	4	5
Find new method inserted	Soufflé	50	50	50	50	50
	Neo4j	108	11	8	6	5

The benchmark results show that:

- Querying engine of **differential factbase** has **constant** query execution times.
- For the **1st** query, **Neo4j** database has **slower** query execution times.
- For the **2nd** query onwards, **Neo4j** database has **faster** query execution times.