Comparative analysis of Relational and Graph databases

Garima Jaiswal, Arun Prakash Agrawal
Lecturer, Information Technology, HCST, Mathura
Assistant Professor, Computer Science, Amity University, Noida

ABSTRACT
Relational database comprises of rows (tuples) and columns (attributes) which is basically used for the storage of data. SQL is used to query the RDBMS which generates the specific results according to the needs of the user. Apart from querying the data, we can also update and modify the records in the database. But unfortunately it is not so easy to perform the above mentioned operations. This is because RDBMS is basically a concept of fixed schema, making changes in one relation may result in multiple changes. Moreover, sometimes for retrieving the results, it is necessary to join multiple relations after satisfying all the constraints that the relations hold. It is difficult to join multiple relations when the size of the data grows. The solution for the above problems is to move on to graph databases. This paper compares the graph databases Neo4j with relational database Oracle 10g.

Keywords
Maturity, Flexibility, Ease of Programming, Retrieval

1. INTRODUCTION
Data is the information that is collected and recorded for retrieving meaningful results. Its essence can easily be implied from it. A database is an assembly of meaningful information tabulated in such a way, that we can easily and quickly retrieve the appropriate result. The database management system is a general purpose software system that promotes the process of defining, constructing and manipulating database for various applications [1]. The relational data model (RDBMS) was first introduced by Ted Cod of IBM Research in 1970 [1]. RDBMS stores data in the form of tables (relation), which consists of rows (tuples) and columns (attributes). This is basically a concept of fixed schema.

RDBMS supports ACID properties: Atomicity, Consistency, Isolation and Durability [2]. Atomicity states either all the operations of a transaction are executed completely or none [12]. Consistency is guaranteed when the transaction on completion takes a database from one consistent state to another [3]. Isolation stated that the execution of a transaction should not be interfered with any other transaction executing concurrently [3]. Durability means that updates to the database must be permanent that is in case of any failure, updates made to database must not be lost [3]. RDBMS usually run adhoc queries. Adhoc query is created to get the information depending upon the necessity of the result. These are not the queries that are predefined or routinely processed. People mostly use RDBMS for data safety, concurrent access, fault tolerance, data integrity, scalability and easy report generation.

Yet, RDBMS is not the best solution. Database schemas are very strict. When the data gets big, then the traditional SQL join operations may not work. Changes made to a single column will resound and would result in multiple changes. And finally, we need look into the mismatch occurring in different tables due to the changes done. With the intense demand to store huge volumes of information, it was a need to switch from relational databases to graph database.

A. Graph Databases
Graph database is a repository in which not only linking among objects is important but also the objects [4]. In a graph database, the objects (entities) are represented by nodes and
the relationship between them is represented by edges. Both the nodes and edges contain the properties [9]. The objects are represented by nodes and the edges represent the relationship between the nodes. The nodes and edges both contain many properties that illustrate their particular characteristics. Graph databases examples are: Allegro Graph, Hypergraph DB, GraphBase, Bitsy and Neo4j [5]. Among all of them only Neo4j is examined here.

B. Neo4j

Neo4j is an open-source graph database supported by Neo Technology. [11]. Neo4j stores data in nodes connected by directed, typed relationships with properties on both, also known as a Property Graph [6]. It was developed by Neo technology. Initially released in 2007. Neo4j version 1.0 was released in February, 2010. Neo4j version 2.0 was released in December, 2013 [7].

Main features of Neo4j are [8]:
- Reliable, with full ACID transactions.
- Highly expandable.
- Easily understandable.
- Rapid and fast retrieval of query result.
- Focuses on what to retrieve, not on how to retrieve.

Cypher is a declarative graph query language that enables demonstrative and productive querying and updating of the graph database [10]. Cypher is a self explanatory and robust language. Cypher is a language that expresses the logic of a computation without describing its control flow. Cypher focuses on what result is to retrieve from graph, not on how to retrieve it [6]. Advantage of this feature is that it promotes the user to target only on their work area.

2. COMPARISION PARAMETRS FOR RELATIONAL AND GRAPH DATABASE.

The assessment between SQL and Cypher is done on different evaluation parameters [9]. On the basis of these evaluation parameters a decision is taken on the selection of the database.

A. View of the Result Set

View of the result set means that how the result of the queries is represented to the user. In SQL the result set is available only in tabular form. No Graph view of the result set is available.

While in Cypher query language, the result set can be viewed in both graphs as well as in table form. We can directly view the properties of all nodes in a graph database but this is not possible in relational database.

B. Complex Search and Join Operations

In SQL to retrieve the result of complex queries, it searches the entire relation. This requires joining of multiple relations. To join a relation referential integrity constraint must be satisfied, which is very time consuming.

While in Cypher query language, to retrieve the result of complex queries it does not traverse the entire graph. It simply checks for all the nodes that have direct relationship with the current node that satisfies the given condition in the query.

C. Retrieval Time

In SQL the retrieval time for the queries is represented in seconds.

While in Cypher query language the retrieval time comes out in milliseconds. Retrieval time is directly proportional to the complexity and number of relations involved in the query [14].

3. IMPLEMENTATION DETAILS

For comparing the SQL and Cypher query language, a query set (Query 1 to Query 4) has been defined.

SQL (Structured Query Language) was used to execute queries on Oracle 11g. Oracle 11g supports relational databases.

Cypher Query Language was used to execute queries on Neo4j Community version 2.0.3, which supports graph databases.

A predefined set of simple and complex queries was defined to compare the execution time of both databases.
In relational database following relations were made with the given schema:
1) Movies: Title, Released, Tagline
2) Actors: Name, Born
3) Relation: Title, Relation_Type, Name

The four queries defined were:
Query 1 (Q1): List all the movies in which Tom Hanks acted.
Query 2 (Q2): Who worked in a movie with tagline everything is connected.
Query 4 (Q4): Who directed the movie tilted speed racer.

4. EXPERIMENTAL RESULTS
After executing the queries the retrieval time was noted down. The outcome has been enumerated in table 1. It can be perceived that the query execution time of Cypher is less than that of SQL. This reason behind it is the search criteria. The relational database evaluates all the records in relation to yield appropriate result that fulfills the given search condition. But, it is very easy to search in graph databases, as they do not evaluate the entire graph. They only check the nodes that have direct relationship to other nodes.

<table>
<thead>
<tr>
<th></th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cypher</td>
<td>216</td>
<td>183</td>
<td>213</td>
<td>226</td>
</tr>
<tr>
<td>SQL</td>
<td>170</td>
<td>200</td>
<td>200</td>
<td>240</td>
</tr>
</tbody>
</table>

Table 1: Query Results In Milliseconds

5. CONCLUSION
The graph databases and relational database both performed well. In general, graph databases accomplished the work more appropriately. When only one relation was involved (as in Query 1 and Query 4) the execution time of both the Cypher and SQL was almost same. But when multiple relations were involved (as in Query 2 and Query 3) the execution time of both the Cypher and SQL varied a lot. It was analyzed that when multiple joins or nested queries or multiple relations were involved, Cypher retrieved the results faster as compared to SQL. Moreover, graph databases are more extensible than relational databases as it is easy to make changes to the existing schema. Due to the divergence in the query execution time of SQL and Neo4j, Neo4j can be used for chemical and biological, enterprise data and in social networks [13].

6. REFERENCES