BUILDING A GRAPH BASED RDF STORE TO MANIPULATE RDF DATA EFFICIENTLY

Ravindra Sampath Ranwala

138227T

Degree of Master of Science/Master of Engineering

Department Computer Science and Engineering

University of Moratuwa

Sri Lanka

March 2015

BUILDING A GRAPH BASED RDF STORE TO MANIPULATE RDF DATA EFFICIENTLY

Ravindra Sampath Ranwala

138227T

Thesis/Dissertation submitted in partial fulfillment of the requirements for the degree Master of Science/Master of Engineering in Computer Science and Engineering

Department of Computer Science and Engineering

University of Moratuwa

Sri Lanka

March 2015

DECLARATION

I declare that this is my own work and this thesis/dissertation 2 does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date:

The above candidate has carried out research for the Masters/MPhil/PhD thesis/Dissertation under my supervision.

Signature of the supervisor: Date:

ABSTRACT

Due to the expansion of semantic web technologies, Resource Description frameworks (RDFs) and triple stores became more prevalent. Since there is a huge amount of RDF data available, managing them in a proper and efficient manner is challenging. Many Triple stores were implemented to support the queries related to semantic web. The queries submitted in this context is called as SPARQL queries which are read dominant. These SPARQL queries needs to be answered quickly and efficiently. RDF data is stored in <subject, predicate, object> form and which is called as a triple. A typical triple store contains billions of triples in the above form.

Much work has been devoted to handle RDF data efficiently. But state of the art systems still cannot handle web scale RDF data effectively. Most existing systems store and index data in particular ways. For an example some systems uses relational tables, bitmap matrix to optimize SPARQL query processing on RDF data. This relational approach suffers from high Join cost and large intermediate results. Some have used prolog inference engine to handle RDF data. This also have some limitations given a huge amount of RDF data.

A modern approach is to model the RDF data in its native Graph form. This approach requires new algorithms to build the graph and graph exploration techniques to answer SPARQL queries. This yields no join cost and very small intermediary results. Also this approach yields less query execution time for complex SPARQL queries.

The objective of this research is to build a graph based triple store for Apache Cassandra. It uses Apache Jena Graph Processing framework to build and explore the RDF graph. Towards the end, it conducts a performance benchmark of this RDF store with some other RDF store implementations using DBPedia dataset and sample queries and proves that this graph based approach outperforms other RDF store implementations.

ACKNOWLEDGEMENT

I would like to express my special appreciation and thanks to my supervisor Dr. Amal Shehan Perera, you have been a tremendous mentor to me. I would like to thank you for guiding me through your experience to make this research more worthwhile. Also I would like to appreciate the extended support and guidance given by Dr. Srinath Perera through his experience and knowledge, which was really helpful for me to complete this successfully.

I would also like to take this opportunity to thank Dr. Malaka Walpola for guiding us towards this work throughout Research Seminar lecture sessions over a year and for the extended support and kindness granted to us. At last but not the least, I would prefer thank to all the academic staff members for helping, guiding, encouraging us and disseminating knowledge throughout the programme.

TABLE OF CONTENTS

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	Description
CQL	Cassandra Query Language
DBMS	Database Management Systems
RDBMS	Relational Database Management Systems
RDF	Resource Description Framework
SPARQL	RDF Query language