Proficient and Scalable RDF Data Management in DiploCloud

Megha Rani Raigonda, Sumayya Nishath

***Abstract*— In this we describe Cloud Computing which concentrates on storage, security and safety of the cloud. where Previously classical techniques where used i.e, Sharding such data using classical techniques lead to insecure data and loss of data because of single cloud where there is a risk of modification of data or the cloud crash or partitioning the graph using traditional min-cut algorithms leads to very inefficient distributed operations and to a high number of joins. In order to overcome problem A new technique is used to retrieve the data from the cloud through duplication of cloud in order to secure the data which cannot be modified in the cloud, then with the help of Diplocloud cloud we are clustering the file to generate an Index through Diplocloud. Here, we describe Proficient and scalable RDF data management in DiploCloud system for the cloud. Contrary to previous approaches. Our system depends on three key strucutres; RDF particle clusters, design records with a successful key document in dual clouds. It routinely concentrate information from a web information source with less measure of time.**

***Index Terms*— Bigdata, Cloud Computing, DiploCloud, RDF, Triple Store.**

I. INTRODUCTION

Cloud computing is a recent trend in IT that moves computing and data away from desktop and portable PCs into large data centers. It refers to applications delivered as services over the Internet as well as to the actual cloud infrastructure [1] with the recent advances in cloud computing which typically contains a significant pool of resources which could be reallocated to different purposes within a short time frames. In addition to this we describe Diplocloud to secure the data safely for the future use and the data can be distributed easily [2],[3]. As in a single node RDF system the data cannot be partitioned easily[4] and it cannot be modified and retrieved by others. For storing the data and implement in-memory RDF engine in graph form leads to bidirectional graph [5], to overcome we form a key index which can have similar data and can be found easily. In this a storage model is presented which proficiently and effectively partitions RDF graph and physically co-locates related instance data In [6], research proposes the most prominent standards are Resource

Description Framework (RDF) and SPARQL Protocol and RDF Query Language (SPARQL). RDF is the standard for storing and representing data and SPARQL is a query language to retrieve data from an RDF store.

.

II. RELATED WORK

[1] Marios D Dikaiakos ET AL.: The key driving forces behind cloud computing are the ubiquity of broadband and wireless networking, falling storage costs, and progressive improvements in Internet computing software. [2] K. Aberer, the grid vine system was one of the first system to context large-scale decentralized RDF management which proficiently partitions and effectively load balancing [3] P. Cudr_e-Mauroux, this genereates much inter-process traffic, with the related triples which can be end up being scattered on all machines [4] K. Zeng, This paper introduces a novel database system for RDF data management called dipLODocus [RDF] , which supports both transactional and analytical queries efficiently. DipLODocus [RDF] takes advantage of a new hybrid storage model for RDF data based on recurring graph patterns. [5] M. Wylot, It builds on trinity a key value store and implement RDF engine memory the data is stored in the form of graph which has adjacency lists for a subject and form a bidirectional subgraph. Then the graph are partitioned [6] Mohammad *Husain this* proposes the Resource Description Framework (RDF). Semantic web technologies can be utilized to build efficient and scalable systems for Cloud Computing. With the explosion of semantic web technologies, large RDF graphs are common place. This poses significant challenges for the storage and retrieval of RDF graphs.

III. EXISTING SYSTEM

The complexities of scaling out an application very much depend on the amount of data process to be. Scaled often the task at hand can be easily split into a large series of subtask to run independently and concurrently such operations are embarrassingly parallel many process are difficult to parallelize typically because they consist of sequential process

Drawbacks- This has difficulty to process unstructured or semistructured data, Web source are continuously evolving results into unpredictable results.

IV. PROPOSED SYSTEM

It uses a non-relational storage format, where semantically related data patterns are mined both from the instance-level and the schema-level data and get co-located to minimize internode operations. A new hybrid storage model that

proficiently and effectively partitions an RDF graph and physically co-locates related instance data.

Advantages- Low computation require for hashing, template list, key index. Trinity does not require training set for processing of web data. Routinely extract data from a web data source with less amount of time.

V. METHODOLOGY

RDF molecule clusters consist of template lists (storing literals in compact lists as in a column-oriented database system) and an efficient key index indexing URIs and literals based on the clusters they belong to. Contrary to many distributed systems, uses a resolutely non-relational storage format.

For Data Partitioning the two standard relational partitioning techniques are round-robin and range partitioning

Data allocation techniques are

Round-Robin: The round-robin allocation simply takes each new molecule it defines and assigns it to the next worker ,Coarse allocation: It splits the incoming data in W parts,Semantic Co-location: The third allocation tries to achieve a tradeoff between load-balancing and co-location

VI. MODULES

We have three main modules

1 Cloud

2 Diplocloud

3 Data user

Module Description

- Cloud

  This collects and store the data in the encrypted so that no other user can modify the data for the security purpose the data is saved first in the cloud then in the Diplocloud. In case the data will be lost in the Diplocloud the data is present in the cloud

- Diplocloud

  In this the user can retrieve the data in two ways by key indexing or by searching template list while retrieving user will get the key to their respective mail-id by using that key the file can be retrieved.

- DataUser

  User uploads and extracts the data from this module with the help of key indexing or template list. For the security of both user and data the cloud server is provided a secret key to the user by using that key the user should login into the cloud .

\

VII. PERFORMANCE ANALYSIS RESULT

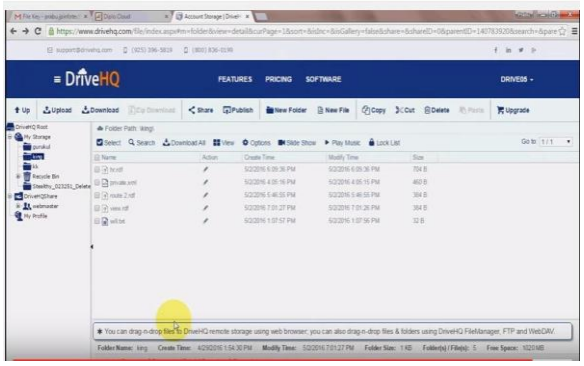

Figure.1. Uploaded File in Cloud

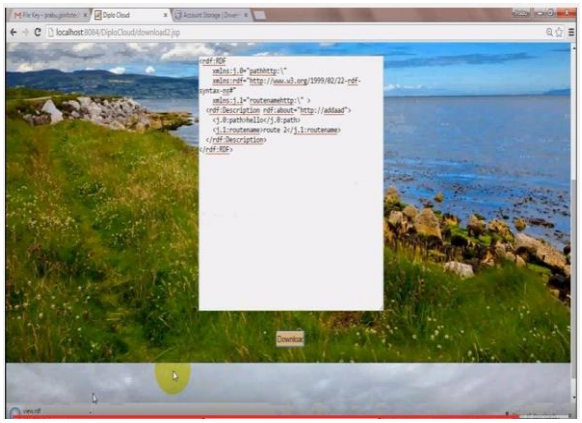

Figure.2. File in Encrypted Form

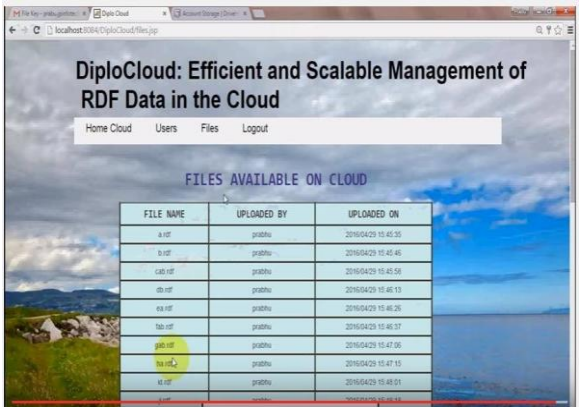

Figure.3. Uploaded Files in DiploCloud

Figure.4. File Available For Download in RDF Form

VIII. CONCLUSION

It strikes an optimal balance between intra-operator parallelism and data co-locating the storage model proficiently partitions the data and effectively co-locate the data. After storing of file in the cloud it clusters the file and generates a key index and it automatically clusters the data from both at schema level and instance level.

REFERENCES

[1] Marios D Dikaiakos, Dimitrios Katsaros, Pankaj Mehra, George Pallis, Athena Vakali," Distributed Internet Computing for IT and Scientific Research," IEEE Internet Comput., vol. 13, no. 5, Sep. 2009.

[2] K. Aberer, P. Cudre-Mauroux, M. Hauswirth, and T. van Pelt,"GridVine: Building Internet-scale semantic overlay networks," inProc. Int. Semantic Web Conf., 2004, pp. 107–121.

[3] P. Cudr_e-Mauroux, S. Agarwal, and K. Aberer, "GridVine: Aninfrastructure for peer information management," IEEE Internet Comput., vol. 11, no. 5, pp. 36–44, Sep./Oct. 2007.

[4] M. Wylot, J. Pont, M. Wisniewski, and P.Cudr_e-Mauroux.(2011). dipLODocus[RDF]: Short and long-tail RDF analytics for massive webs of data. Proc. 10th Int. Conf. Semantic Web - Vol. Part I, pp. 778–793 [Online]. Available: http://dl.acm.org/citation.cfm? id=2063016.2063066

[5] K. Zeng, J. Yang, H. Wang, B. Shao, and Z. Wang, "A distributed graph engine for web scale RDF data," Proc. VLDB Endowment, vol. 6,no. 4, pp 265–276,2013.

[6] Mohammad Husain, James McGlothlin, Mohammad M Masud, Latifur Khan, Bhavani M Thuraisingham, "Heuristics-Based Query Processing for Large RDF Graphs Using Cloud Computing," IEEE Internet Comput., vol. 23, no. 9, pp. 1312–1327, Sep. 2011.

Ms. Megha Rani Raigonda working as Assistant Professor Department of Studies in Computer Application(MCA), Visvesvaraya Technological University Centre for PG Studies Kalaburagi, India. M.Tech Computer Science and Engineering, pursuing PhD. Published 3 journal paper, attended and presented paper in National and International Conference. Mobile No +918088681632

.

Ms. Sumayya Nishath working as student, Department of studies in Computer Application(MCA), Visvesvaraya Technological University Centre for PG Studies Kalaburagi, India, Mobile No +918867036350