IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-6, Special Issue-2, March 2018 E-ISSN: 2347-2693

Efficient Storage and Accessing Through Query Process

M. Kalaivani[1], J.Jayanthi[2,] S.Hareesh[3]

[1]Department of Computer Science, Srimad Andavan Arts and Science College (Autonomous), Trichy
[2,3]Department of Computer Science, Srimad Andavan Arts and Science College (Autonomous), Trichy

Available online at: www.ijcseonline.org

***Abstract*-**The explosion of varied Linked information on the Web poses new challenges to file systems. In exacting, the ability to store, track, and query origin data is flattering a essential characteristic of modern triple stores. Here, present methods extending a inhabitant RDF accumulate to professionally handle the storage, tracking, and querying of provenance in RDF data. Here, explain a dependable and comprehensible requirement of the way consequences was resultant from the information and how particular pieces of data were combined to answer a query. Subsequently, there techniques to mold queries with attribution data. To empirically judge the accessible methods and reveal that the transparency of storing and track attribution is suitable. Finally, show that dressmaking a query with origin information can also considerably get better the presentation of inquiry execution.

I. INTRODUCTION

A central use-case for reserve Description Framework (RDF) data organization systems is data addition. Data is acquired from manifold sources either as RDF or rehabilitated to RDF; schemas are mapped. There are a diversity of systems such as fate and the connected Data Integration Framework that realize this integration process. The heterogeneity of data shared with the ability to without problems integrate it— using values such as RDF and SPARQL—mean so as to the support of origin within these system is a key feature. For case a user may covet to trace which sources were instrumental in only if results, how data sources were combined, to validate or invalidate results, or to adapt queries specifically based on provenance in order. Within the Web community, there have been quite a few efforts in mounting models and syntaxes to spell out and trace attribution, which resulted in the recent W3C PROV advice.

However, less notice has been given to the efficient handling of such provenance data in RDF database systems. The most common mechanism used within RDF data management is named graph. This mechanism was recently standardized. Named graphs associate a set of triples with an URI.Thus, while RDF databases, i.e., triplestores, support named graphs, there has only been a relatively small number of approaches specifically focused on provenance within the triplestore itself and much of it has been focused on theoretical aspects of the problem rather than efficient implementations. Given the prevalence of provenance in Web Data—36% of datasets contain provenance data and the use of named graphs this article shows how RDF databases can effectively track the ancestry of queries and perform query that invent from data scoped with provenance in sequence (i.e., provenance-enabled queries). In the following, we present TripleProv, a new database system supporting the transparent and automatic derivation of detailed provenance information for arbitrary queries and the execution of queries with respect to provenance data. TripleProv is based on a subject RDF store, which we have extensive with two dissimilar attribution conscious storage models and co-location strategy to accumulate provenance data in a compact fashion. In addition, TripleProv supports query execution strategies to derive provenance information at two different levels of granularity and to scope queries with provenance information. The contribution of the work presented in this article is the integration of our previous approaches into provenance enabled triple store. The new version of the system allows the user to execute provenance-enabled queries and at the same time obtain a provenance polynomial of the query results. Moreover, we there two new untried scenarios thoroughly evaluate the scalability of our techniques.

II. RELATED WORK

[1] Cloud service offerings such as Amazon's Simple Storage Service (S3), a part of Amazon Web Services, offer cheap storage at a fixed cost per gigabyte (no minimums or maximums) and are appealing for backup, since they provide an easy way to safely store data off-site. There are pre-packaged online services specifically built for backup, such as Mozy and Carbonite. Cumulus explores the other end of the intend space: construction on pinnacle of a very generic cloud storage layer, an example of what we refer to as building on the "thin cloud." Using a generic, minimalist interface means that Cumulus is portable to virtually any online storage service the client implements all request logic. Cumulus is not sole in this continue, but compare with obtainable support tackle target S3, Cumulus achieve lower costs, showing that this

incomplete boundary is not an impediment to achieving a very low and competitive cost for backup.

[2] At the same time as the earth moves to digital storage space for archival purposes, there is an growing demand for systems that can offer secure data storage space in a cost-effective manner. By identifying ordinary chunks of data together within and between records and storing them only once, can yield cost investments by growing the utility of a given quantity of storage. Unfortunately, exploits identical content, while encryption attempts to make all content appear random; the similar content encrypted by means of two dissimilar keys consequences in very unlike ciphertext. Thus, combining the space competence of with the confidentiality aspects of encryption is problematic. We have residential a solution that provides both data safety and space competence in single-server luggage compartment and dispersed storage systems. Encryption keys are generated in a reliable manner from the chunk data; thus, the same chunks will forever encrypt to the same ciphertext.

[3] The Farsite dispersed file system provides ease of use by replicating each file onto numerous desktop computers. Since this duplication consumes important storage space, it is significant to get back used freedom where possible. dimension of over 500 desktop file systems shows that nearly semi of all inspired space is busy by files. To present a method to reclaim space from this incidental to make it obtainable for forbidden file replication. This mechanism includes convergent encryption, which enables files to coalesced into the space of a lone file, even if the files are encrypted with dissimilar users' keys, and a Self-Arranging, Lossy, Associative Database for aggregate file content and site information in a decentralized, scalable, fault-tolerant manner.

[4] File systems hosting virtual machines characteristically hold many blocks of data resulting in wasted storage space and increased storage array cache footprint. This addresses these problems by storing a single example of each exclusive data block and distribution it between all original sources of that data. To suggest DEDE, a block-level method for live cluster heading system that does not demand any middle coordination, tolerates crowd failures, and takes advantage of the block layout policies of an cluster file system. In hosts keep summaries of their own writes to the come together file scheme in shared on-disk logs. Each host periodically and separately processes the summaries of its locked files, merges them with a communal index of blocks, and reclaims any photocopy blocks. This manipulates metadata using universal file system interfaces without information of the file system implementation.

[5] Consider architecture for a server less dispersed file scheme that does not suppose mutual trust amongst the client computers. The system provides security, ease of use, and dependability by distributing multiple encrypted replicas of each file amongst the client machines. To charge the viability of deploying this system on an obtainable desktop infrastructure, gauge and analyze a huge set of client machines in a profitable environment. In particular, calculate and report consequences on disk usage and content, file activity; and mechanism uptimes, lifetimes, and loads. The conclude that the deliberate desktop infrastructure would sufficiently hold up our proposed system, as long as availability on the order of one empty file ask for per user per thousand days.

III. METHODOLOGY

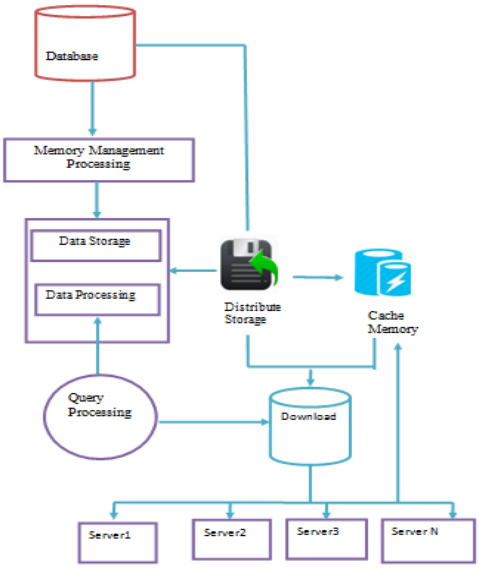
EXISTING PROCESS

In our existing, Disk based system is used for responding due to high access latency to hard disks. Multi-core processors and the availability of large amounts of main memory at plummeting cost are creating new breakthroughs, creation of it feasible to build in-memory systems where a major part of the folder fits in memory.

PROPOSED PROCESS

Here, DRAM, Distributed database, relational database are used to manage the memory efficiently. In-memory data management and processing proposals and systems, including Both data storage systems and data processing frameworks. To improve performance, including reducing dependency on the hard disk, adding more memory to make more data resident in the memory, and even deploying an in-memory system where all data can be kept in memory

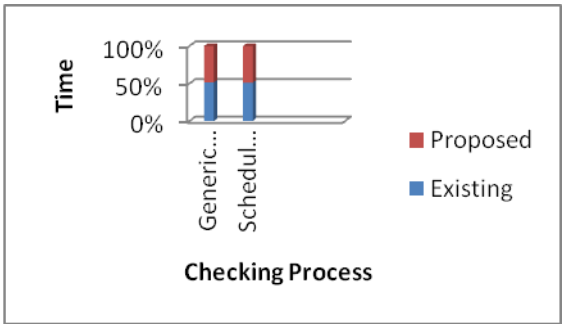
ARCHITECTURE



IV. CONCLUSION

To the best of our knowledge, obtainable the first effort to translate hypothetical approaching from the file provenance literature into a high-performance triple store. This technique allow not only easy tracing of ancestry for inquiry results, but also considers fine-grained multilevel provenance and permit us to tailor the query implementation with origin information. This introduced two storage space models and five query implementation strategies for behind attribution in Linked Data organization systems. From our investigate can say that the extra data have to course the slower queries are executed and the more discerning is the ascription query the more enlarge in performance. Less discriminatory workload queries are more responsive to those aspects than queries that authorize to early prune middle results. The more higher query execution strategies that take benefit of the selectivity of the attribution information are more sensitive to the number elements returned by the provenance query.

PERFORMANCE ANALYSIS



REFERENCES

[1] T. Heath and C. Bizer, Linked Data: Evolving the Web into a Global Data Space, ser. Synthesis Lectures on the Semantic Web. Morgan & Claypool Publishers, 2011.

[2] C. A. Knoblock, P. Szekely, J. L. Ambite, S. Gupta, A. Goel, M. Muslea, K. Lerman, M. Taheriyan, and P. Mallick, "Semi-automatically mapping structured sources into the semantic web," in Proceedings of the Extended Semantic Web Conference, Crete, Greece, 2012.

[3] A. Schultz, A. Matteini, R. Isele, C. Bizer, and C. Becker, "Ldif - linked data integration framework," in COLD, 2011.

[4] P. Groth and L. Moreau (eds.), "PROV-Overview. An Overview of the PROV Family of Documents," World Wide Web Consortium, W3C Working Group Note NOTE-prov-overview-20130430, Apr. 2013.

[5] J. J. Carroll, C. Bizer, P. Hayes, and P. Stickler, "Named graphs, provenance and trust," in Proceedings of the 14th International Conference on World Wide Web, ser. WWW '05. New York, NY, USA: ACM, 2005, pp. 613–622.

[6] D. W. R. Cyganiak and M. L. (Ed.), "RDF 1.1 Concepts and Abstract Syntax," W3C Recommendation, February 2014, http://www.w3.org/TR/rdf11-concepts/.

[7] M. Schmachtenberg, C. Bizer, and H. Paulheim, "Adoption of the linked data best practices in di_erent topical domains," in The Semantic Web, ISWC 2014, ser. Lecture Notes in Computer Science, P. Mika, T. Tudorache, A. Bernstein, C. Welty, C. Knoblock, D. Vrandei, P. Groth, N. Noy, K. Janowicz, and C. Goble, Eds. Springer International Publishing, 2014, vol. 8796, pp. 245–260. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-11964-9 16

[8] G. Grimnes. (2012) BTC2012 Stats. [Online]. Available: http: //gromgull.net/blog/2012/07/some-basic-btc2012-stats/

[9] P. Groth and W. Beek. (2016) Measuring PROV Provenance on the Web of Data. [Online]. Available: https://nbviewer.jupyter.org/github/pgroth/prov-wod-analysis/blob/master/MeasuringPROVProvenanceWebofData.ipynb

[10] M. Wylot, J. Pont, M. Wisniewski, and P. Cudr´e-Mauroux, "dipLODocus[RDF]: short and long-tail RDF analytics for massive webs of data," in Proceedings of the 10th international conference on The semantic web - Volume Part I, ser. ISWC'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 778–793. [Online]. Available: http: //dl.acm.org/citation.cfm?id=2063016.2063066

[11] M. Wylot and P. Cudr´e-Mauroux, "DiploCloud: E_cient and Scalable Management of RDF Data in the Cloud," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 3, pp. 659–674, March 2016.

[12] M. Wylot, P. Cudre-Mauroux, and P. Groth, "TripleProv: E_cient Processing of Lineage Queries in a Native RDF Store," in Proceedings of the 23rd International Conference on World Wide Web, ser. WWW '14. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2014, pp. 455–466.

[13] M. Wylot, P. Cudr´e-Mauroux, and P. Groth, "Executing Provenance-Enabled Queries over Web Data," in Proceedings of the 24rd International Conference on World Wide Web, ser. WWW '15. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2015.

[14] "A demonstration of TripleProv: tracking and querying provenance over web data," vol. 8, no. 12. VLDB Endowment, 2015, pp. 1992–1995.

[15] L. Moreau, "The foundations for provenance on the web," Foundations and Trends in Web Science, vol. 2, no. 2–3, pp. 99–241, Nov. 2010. [Online]. Available: http://eprints.ecs.soton.ac.uk/21691/

