An Efficient and Scalable RDF Indexing Strategy based on B-Hashed-Bitmap Algorithm using CUDA

International Journal of Computer Applications
© 2014 by IJCA Journal

Volume 104 - Number 7
Year of Publication: 2014

Authors:
Sharmi Sankar
Munesh Singh
Awny Sayed
Jihad Alkhalaf Bani-younis

10.5120/18216-9221
{bibtex}pxc3899221.bib{/bibtex}

Abstract

Indexing enormous databases such as RDF has been a focus of intense research. As is well understood, indexing plays a pivotal role in speeding up data retrieval operations and query performance. Besides expediting search, an index can motivate new data-store schemes and technologies that can possibly revolutionize large data-analytics engine design, more often relevant to semantic web. Due to the proliferation of internet and the ease of creating and generating data on the fly - handling, storing and the subsequent semantic processing has proven to be a major bottleneck for the RDF data community. Handling data of such scale and magnitude requires a parallel approach as provided by the GPUs (Graphical processing units). In this paper, a new efficient and scalable index is proposed that uses a combination of B+ trees, hashing and sparse matrices. These data structures have an edge over others in terms
of their implementation as a parallel algorithm using the CUDA (Compute Unified Device Architecture) framework meant to program massively parallel GPU multicores. So far, RDF data has been mostly implemented either as a RDBMS or as a non-native data-store, in both cases the sequential indexing strategy fails miserably with the scaling of the data-store. Parallel implementation of indices provides a suitable option for dealing with scalable and dynamically generated data over distributed networks. The crucial sparse matrix part of the proposed index is benchmarked against different CUDA memory implementations to derive optimal matrix processing options. The sparse matrix search is profiled using cudamemchk and visual profiler for identifying bottlenecks and inconsistencies in thread execution called thread divergence. Benchmarking the data provides promising results for a B+ tree based index coupled with hashing and sparse matrix implementations.

References

- Wolfgang Nejdl, Hadhami Dhraief, Martin Wolpers, O-Telos-RDF: A Resource Description Format with Enhanced Meta-Modeling Functionalities based on O-Telos
- Speeding up on-disk RDF index lookups using B+Hash trees, Minh Khoa Nguyen, Cosmin Basca, Abraham Bernstein, IOS Press, 2012
- Hexastore: Sextuple Indexing for Semantic Web Data Management, Cathrin Weiss, Panagiotis Karras, Abraham Bernstein
- Large RDF Representation Framework for GPUs Case Study Key-Value Storage and Binary Triple Pattern, Chidchanok Choksuchat, Chantana Chantapornchai, International Computer Science and Engineering Conference (ICSEC), 2013
- Optimizing RDF stores by coupling General-purpose Graphics Processing Units and Central Processing Units, Bassem Makni
- Erling and Mikhailov, RDF Support in the Virtuoso DBMS
Index Terms

Computer Science

Keywords
RDF  B+ tree  hashmap  sparse matrix  CUDA  GPU.