A Comparison of Different Approaches of Distributed Data Processing

Tigran Shahinyan

Institute for Informatics and
Automation Problems
Yerevan, Armenia
e-mail: Tigran.Shahinyan@gmail.com

ABSTRACT

The emergence of multiple new sources of manually and automatically generated data, e.g. cell phones, sensors, social networks, etc., spawns huge amounts of data which is hard to process effectively using conventional approaches. Several new approaches have recently been introduces to manage efficient distributed processing of such data. No solution is universal and applicable for all cases, hence a comparative analysis and benchmarking of different approaches are necessary. The presented work is an attempt to compare the efficiency of using relational, non-relational (MapReduce), and hybrid approaches of distributed data processing.

Keywords

Parallel computing, relational databases, grid computing, mapreduce.

1. INTRODUCTION

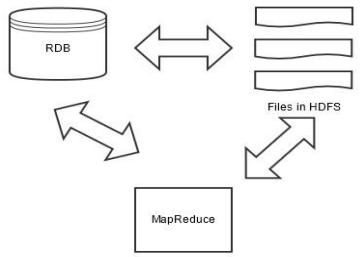
The global network is becoming an enormous consolidation of huge amounts of data of different origins and representations. The effective usage of this data is becoming a challenge. Different nature of data dictates different approaches of processing. Our goal is to compare different approaches of distributed data processing depending on different data structures.

2. Data Models

Relational database model is a model for representing and processing data in relations using the first-order predicate logic. The data is organized in tuples and tables supporting the relational calculus operations. [1]
Modern relational database systems support flavors of a declarative query language SQL. Most of the relational database system support ACID (Atomicity, Consistency, Isolation, and Durability) transactions guaranteeing data safety. [2]
As an alternative to the relational database model several types of semi-structured database models are used. Document-oriented databases are used to store, retrieve, and manage document-oriented information. XML databases use tree structures for representing data and recursive processing (XPath and XQuery) for data retrieval. NoSQL databases – often use key-value pair representation of data, which is easier to distribute horizontally. NoSQL databases are easier to scale-out but the tradeoff is no full support of ACID transactions.
MapReduce is a paradigm introduced by Google for processing huge datasets on certain kinds of distributable problems using a large number of computers (nodes), a cluster or a grid. It is based on two functions: Mapper and Reducer, which are used as arguments by higher-order functions Map and Reduce [3].
A commonly used implementation of MapReduce is Apache Hadoop which among other features supports a distributed file system HDFS. HDFS is a distributed virtual file system used by Hadoop to storing input and output data with high durability.

3. Distributed Data Processing Approaches

The most common and obtainable architecture for distributed relational database system is shared nothing architecture. As opposed to the shared memory and the shared disk architectures it's easier to implement on cluster and grid computing infrastructures. As an example of a Distributed Relational Database Systems we've considered MySQL Cluster. [4]
For MapReduce implementation Hadoop framework was used with HDFS file system.
The hybrid approach uses a relational database for processed data and HDFS and MapReduce for storing and processing



Each of the approaches has its strength and weaknesses. MySQL cluster shows best performance for OLTP applications with relatively modest amount of data.
MapReduce fits best for tasks of processing huge amounts of input data,e.g., analyzing daily web-logs, gps or sensor data.
The hybrid approach is the best solution for providing OLTP services based on input data from large amount of active devices.

4. References

[1] E. F. Codd, A relational model of data for large shared data banks, Comm. of the ACM, Vol. 13, Issue 6, 1970

[2] Jim Gray, The Transaction Concept: Virtues and Limitations, Proceedings of the 7th VLDB conf., Sep. 1981

[3] Jeffrey Dean, Sanjay Ghemawat, MapReduce: A Flexible Data Processing Tool, Communications of the ACM, Volume 53 Issue 1, January 2010

[4] MySQL Cluster 7.0 & 7.1: Architecture and New Features, Technical White Paper, Oracle, October 2010