Big Data Paper Summary

Philip Kirwin

10/20/2016

References:

"Bigtable: A Distributed Storage System for Structured Data" by Fay Chang, et. Al.

"A Comparison of Approaches to Large-Scale Data Analysis" by Andrew Pavlo, et. Al.

Michael Stonebreaker's talk on his 10-Year Most Influential Paper Award

What is Bigtable?

- Bigtable is a data storage system designed to store incredibly massive amounts of data on potentially thousands of machines.
- It is created and primarily used by Google for several of its largest products, including Google Earth and Google Analytics.
- It is represented as a multi-dimensional sorted map that is indexed by row key, column key, and timestamp.
- Data in Bigtable is ordered by row key. The row range of a table is partitioned into units of distribution and load balancing called tablets, such that each only needs to communicate with a small number of machines.
- Column keys are stored in column families that are used for access control.
- Timestamps allow multiple versions of the same data to be indexed.

How is Bigtable Implemented?

- Bigtable's API includes functionality for creating or deleting tables and column families as well as changing cluster, table, and column family metadata.
- Bigtable can be used with MapReduce as an input source and output target for large-scale computations.
- Bigtable takes advantage of other Google services, such as Google File System for storing log and data files as well as the Google SSTable file format to store internal data as key/value pairs.
- Systems that implement Bigtable need a library linked to every client, one master server that assigns tablets to tablet servers, and many tablet servers that each manage a set of tablets.

Analysis of Bigtable

- Bigtable seems to provide a rather versatile means of managing the incredibly large amounts of information that products like Google Earth or Google Analytics can generate.
- The organization of data into arbitrary row keys and column keys in addition to timestamps is a very simple and user-friendly storage method that still allows for a great deal of data diversity.
- The implementation of Bogtable may be more complex than most by requiring a library on every client, but the ability to control the operation with a single master that communicates with tablet servers, then tablets, then individual rows seems very efficient.

Comparing MapReduce to Database Management Systems

- Cluster computing paradigms, such as MapReduce, suggest that a large number of low-end processors working in parallel are more efficient in solving computing problems than a smaller set of high-end servers used for most database management systems.
- Database management systems force the use of strict schemas, indexing, and a relational programming model, while MapReduce's simplicity forces the programmers to create these for themselves, which can be difficult if sharing is needed for the system.
- While cluster systems like MapReduce have faster load time when searching for a pattern in a record, database management systems are faster in executing aggregation, joining, and selection tasks.

Implementation of the Large-Scale Data Analysis Comparison

- While MapReduce systems take less effort to set up and thus can save money, they tend to be less efficient than database management systems.
- Database management systems benefit from the wealth of technologies developed for them during their long existence.
- MapReduce does benefit from having a higher tolerance for node failure than database management systems, but needs more nods to match its efficiency, thus opening up more room for failures.
- The solution may be to implement a MapReduce system and an SQL system that work together.

Analysis of the Large-Scale Data Analysis Comparison

- MapReduce systems and database management systems each have their own well-defined strengths, so it is difficult to gauge which of the two is generally better.
- The necessary functionality of a specific data collection would likely be a better indicator of which system would perform better in a specific case.
- While the authors of this paper do present a convincing argument for the continued relevancy of database management systems, this does not mean there are no cases where MapReduce would be a preferred system.

Comparing the Two Papers

- According to the comparison paper, Bigtable, being a user of MapReduce, may see more efficient functionality by utilizing a database management system instead.
- However, the uses of Bigtable include products such as Google Earth and Google Analytics, which store massive amounts of data at a time, meaning that MapReduce's faster loading time and ability to tolerate failure would be very useful.
- Due to the heavy data loads experienced by Bigtable, having a slower load time would much more drastically reduce the overall efficiency of the products it is used in, thus justifying the efficiency lost through MapReduce usage.

Michael Stonebreaker Talk

- In his 2005 paper, "One Size Fits All – An Idea Whose Time Has Come and Gone", Stonebreaker asserts that the relational database management system was a dying medium.
- Ten years later, he is confident that the relational database system is completely dead.
- In the yeas since the original paper, the data storage system market has opened to a large variety of new models that are superior to the relational model in flexibility and speed.
- This means that now is an excellent time to be a database management system researcher; because new possibilities are being discovered constantly.

The Advantages and Disadvantages of Bigtable

- Advantages:
  - Faster loading times for large amounts of data.
  - High tolerance for node failure.
  - Makes use of newer technologies that leave room for innovation.
- Disadvantages:
  - Less efficient task completion.
  - More difficult for outsiders to understand.
  - Requires unique scripts for structure, indexing, etc.