Google BigQuery - A data warehouse for large-scale data analytics

SAGAR VORA[1]

[1] *School of Informatics and Computing, Bloomington, IN 47408, U.S.A.*

March 25, 2017

There has been an increase in the volume of relational data generated today through a large number of sources. The large volume of data forces us to find solutions which can cope with them. Recently several hybrid approaches like HadoopDB, Hive, etc have been introduced to handle this large data. Although these have been successful in handling large data, but their architecture makes them inefficient to handle suboptimal execution strategies. Moreover this data is the information that companies would like to explore and analyse quickly to identify strategic answers to business. Therefore, in order to solve the problem of traditional database management systems to support large volumes of data arises Google's BigQuery platform. This solution runs in cloud, SQL-like queries against massive quantites of data, providing real time insights about the data. In this paper, we will analyze the main features of BigQuery that Google offers to manage large-scale data.

Keywords: BigQuery, ETL, BigData, Google, Cloud, Database, SaaS, SQL

https://github.com/vorasagar7/sp17-i524/paper2/S17-IR-2041/report.pdf

INTRODUCTION

Nowadays, the amount of data being collected, stored and processed continues to grow rapidly. Therefore high performance and scalability have become two essentials requirements for data analytics systems. Querying massive datasets can be time consuming and expensive without the right hardware and infrastructure. Google BigQuery solves this problem by enabling super-fast, SQL-like queries against append-only tables, using the processing power of Google's infrastructure. Google BigQuery [1] [2] is a cloud web service very attractive for its ease of use and functionality. It is ideal for businesses that cannot invest in infrastructure to process a huge amount of information. This platform allows to store and retrieve large amounts of information in near real time with main focus on data analysis.

CLOUD COMPUTING

Cloud computing [3] [4] allows access to computing resources easily scalable and virtualized via the Internet. The use becomes simpler because users need not to have knowledge, experience or management of the infrastructure. There are usually three types of cloud offerings:

- Infrastructure as a service (IaaS) which provides basic compute and storage resources including servers, networking.
- Platform as a service (PaaS) which provides a platform allowing users to develop, run and manage applications without the complexity of building and maintaning the infrastructure typically associated with developing and launching apps.
- Software as a Service (SaaS) which refers to one of the most known cloud services, which consists of applications running directly in the cloud provider.

GOOGLE BIGQUERY

To solve the problems faced by Hadoop [5] MapReduce [6], Google developed Dremel application in order to process large volumes of data. Dremel was designed to deliver high performance on data which was distributed across multiple servers and SQL support. But in 2012 at Google I/O event, it was announced the end of the Dremel and the beginning of BigQuery which became then the high-performance cloud offering of google.

Google BigQuery [2] platform is a SaaS as a model in the cloud. It is not a reporting system and does not have an interface that allows the operation of the information but it is ideal to export results by Tableau, QlikView, Excel, among others including the tools of Business Intelligence (BI) as seen in Figure 1. Projects are top-level containers that store information about billing and authorized users, and they contain BigQuery data. When we create a new project, it is identified by a name, authorized users and data [8].

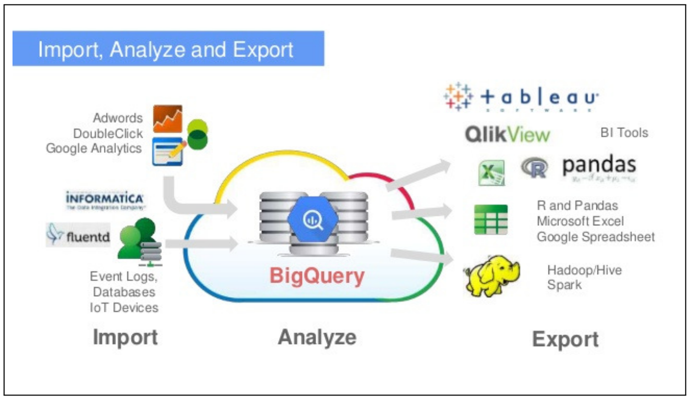

Fig. 1. [7] System Architecture of BigQuery

Data which is generated from a variety of sources like event logs, relational databases, IoT devices like sensors, social media websites, etc is given as an input to BigQuery which processes it to generate some meaningful anaylsis according to the requirement. The final data could be represented and exported using Tableau, Qlikview and other BI tools. It can also be exported on Hadoop system for pararell processing 1.

FEATURES OF BIGQUERY

Google BigQuery [1] presents some characteristics like Velocity, Scalability, Security and multiple access methods.

Velocity

BigQuery can process millions of information which is not indexed in seconds due to columnar storage, and tree architecture as explained below.

Columnar Storage

The data instead of being stored in terms of rows, the data is stored as columnsand thus storage will be oriented. This will result in scanning of only the required values, largely reducing latency. This storage also allows for a higher compression ratio. Google reports [9] that they can achieve columnar compression ratios of 1:10 as opposed to 1:3 when storing data in a traditional row based format. The reason behind this is the similarity of data stored in the columns as the variation is significantly lower than when the data is stored in rows.

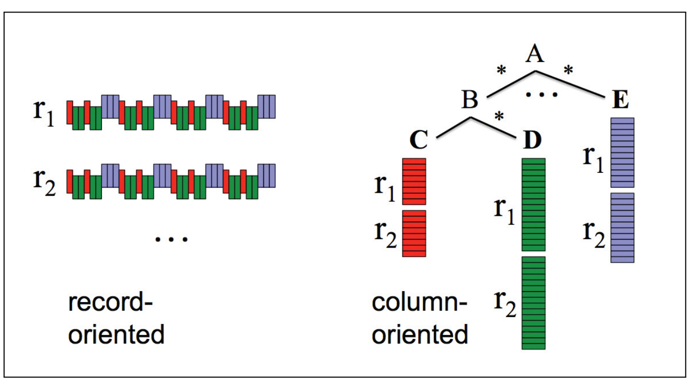

Fig. 2. [9] Row vs Column Storage

Tree Architecture

This is used for processing queries and aggregating results across different nodes. BigQuery is spread across thousands of servers. The data is sharded across multiple machines 2. This helps retrieve data much faster. Let us see this with the help of 2 examples.

Here, A is our root server, with B being an intermediate server and C, D, E being leaf servers having local disk storage.

Example 1: Speed Statement: List out all the customer names starting with 'G'

Let us assume Node C contains customer names. Hence, it is as simple as traversing A -> B -> C and looking at the datasets Cr1 and Cr2 for names starting with G. One need not look at A -> B -> D and A -> E. Hence, in this simple scenario, we have already achieved a search time of 0.33x that of a typical RDBMS (assuming equal column sizes).

Example 2: Parallel Processing Statement: Count all the customer names starting with 'G'

- Root A passes the query to intermediate B.
- B translates the query to Sum (Count all the customer names starting with 'G').
- Intermediate B passes this query to Leaf C.
- Leaf C has tablets (Horizontal sharded tables) r1 and r2.
- C accesses these tablets in parallel, runs quick counts and passes the count of Cr1 and Cr2 to B.
- B sums Cr1 and Cr2 and passes it to the root A for output.

Now, if this architecture was scaled to thousands of leaf servers and hundreds of intermediate servers. This achieves enormous amounts of parallel processing by utilizing a small percentage of the CPU processing and Disk I/O of each server as opposed to 100% usage of fewer servers.

Scalability

It is the ability to manage huge data size with millions of records reaching terabytes of information, without space limits .

Simplicity

BigQuery provides a simple interface to upload and execute browse through a query language similar to SQL 3.

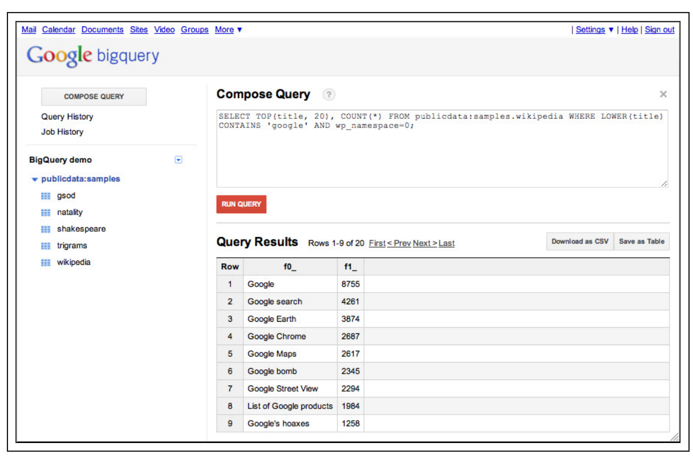

Fig. 3. [10] Big Query User Interface

Multiple permissions

It is the capacity to manage different access permissions, read - only, editing or owner.

Security

To ensure security, the solution makes use of SSL (Secure Sockets Layer)

Multiple access methods

We can access the service in different ways. We can use a BigQuery Browser tool, a bq Command-line tool or a REST-based API.

- BigQuery Browser Tool: With this tool it is possible to easily browse, create tables, run queries, and export data to Google Cloud Storage.
- bq command-line Tool: This Python command - line tool permits to manage and query the data.
- REST API: We can access BigQuery making calls to the REST API using a variety of client libraries such as Java, PHP or Python.

EXPERIMENT

In [**?**]r, the performance of Tajo and Hive has been compared by using 1TB TPC-H benchmark set. Figure **??** shows the experimental results. For this experiment, they have used an in-house cluster of 32 nodes, each of which is equipped with 16GB RAM, 4TB HDD, and an Intel i5 quad core CPU. The x-axis means the TPC-H queries and the y-axis indicates the processing times. The results show that the time take by Tajo to execute SQL queries is less than Apache Hive on the top of MapReduce.

COST

Many prices are practiced, existing free levels. It is charged to the customer by the total information processed but for loading, reading or export information there is no associated cost. There is also budget option for a particular project.

CONCLUSION

In this paper I present Google BigQuery platform, which is a cloud-based database service that is able to process large data sets quickly. BigQuery allows to run SQL-like queries against multiple terabytes of data in a matter of seconds. It is suitable for ad hoc OLAP/BI queries that require results as fast as possible. As a cloud-powered massively parallel query database it provides extremely high full-scan query performance and cost effectiveness compared to traditional data warehouse solutions and appliances.

ACKNOWLEDGEMENTS

I would like to thank my professor Gregor von Laszewski and all the associate instructors for their constant technical support.

REFERENCES

[1] "What is bigquery," Web Page, accessed: 2017-3-24. [Online]. Available: https://cloud.google.com/bigquery/
[2] S. Fernandes and J. Bernardino, "What is bigquery?" in *Proceedings of the 19th International Database Engineering & Applications Symposium*. New York, NY, USA: ACM, 2014, pp. 202–203. [Online]. Available: http://doi.acm.org.proxyiub.uits.iu.edu/10.1145/2790755.2790797
[3] "What is cloud computing," Web Page, accessed: 2017-3-24. [Online]. Available: https://apprenda.com/library/cloud
[4] C. Binnig, D. Kossmann, T. Kraska, and S. Loesing, "How is the weather tomorrow?: Towards a benchmark for the cloud," in *Proceedings of the Second International Workshop on Testing Database Systems*. New York, NY, USA: ACM, 2009, pp. 9:1–9:6. [Online]. Available: http://doi.acm.org/10.1145/1594156.1594168
[5] "Apache hadoop," Web Page, accessed: 2017-3-25. [Online]. Available: http://hadoop.apache.org/
[6] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008, published as an Article. [Online]. Available: http://doi.acm.org/10.1145/1327452.1327492
[7] K. Sato, "Fluentd + google bigquery," Web Page, Mar. 2014, accessed: 2017-3-24. [Online]. Available: https://www.slideshare.net/GoogleCloudPlatformJP/google-for-1600-kpi-fluentd-google-big-query
[8] "What is bigquery? bigquery documentation google cloud platform," Web Page, accessed: 2017-3-23. [Online]. Available: https://cloud.google.com/bigquery/what-is-bigquery
[9] V. Agrawal, "Google bigquery vs mapreduce vs powerdrill," Web Page, accessed: 2017-3-23. [Online]. Available: http://geeksmirage.com/google-bigquery-vs-mapreduce-vs-powerdrill
[10] J.-k. Kwek, "Google bigquery service: Big data analytics at google speed," Blog, Nov. 2011, accessed: 2017-3-23. [Online]. Available: https://cloud.googleblog.com/2011/11/google-bigquery-service-big-data.html