International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

Google Cloud Dataflow – An Insight

Eashani Deorukhkar[1]

Department of Information Technology, RamraoAdik Institute of Technology, Mumbai, India

Abstract: ***The massive explosion of data and the need for its processing has become an area of interest with many companies vying to occupy a significant share of this market. The processing of streaming data i.e. data generated by multiple sources simultaneously and continuously is an ability that some widely used data processing technologies lack. This paper discusses the "Cloud Data Flow" technology offered by Google as a possible solution to this problem and also a few of its overall pros and cons.***

Keywords: Data processing, streaming data, Google, Cloud Dataflow

1. Introduction

The processing of large datasets to generate new insights and relationships from it is quite a common activity in numerous companies today. However, this process is quite difficult and resource intensive even for experts[1]. Traditionally, this processing has been performed on static datasets wherein the information is stored for a while before being processed. This method was not scalable when it came to processing of data streams. The huge amounts of data being generated by streams like real-time data from sensors or from social networks cannot be stored before processing as new information is created continuously. In addition to this, the emergence of cloud computing required data processing applications to be able to seamlessly work in a cloud environment. Project Cloud Dataflow by Google is designed to be able to overcome these obstacles when it comes to large scale data processing. Instead of assuming that finite pools that will eventually be complete, a new approach is to consider them to be infinite and continuously changing.[5] The Dataflow model works on this principle.

2. Existing Technology

One of the most popular data processing frameworks that dominate the market today is Hadoop created by the Apache Software Foundation. While it is designed for the processing of large datasets, it comes with a few disadvantages like:

- Reduction in performance when datasets are in the order of petabytes[2]
- Hadoop can handle batch processing only

Technologies like Apache Storm and Apache Spark are better equipped to handle streaming data.

Cloud Dataflow is Google's answer to these technologies. It can be used for the following:

- For data integration and preparation[2]
- To examine a real-time stream of events for significant patterns and activities[2]
- To implement advanced, multi-step processing pipelines to extract deep insight from datasets of any size[2]

3. Technical aspects

3.1 Overview

This model is specifically intended to make data processing on a large scale easier. It delegates the physical implementation of parallel processing to one of the "Cloud Dataflow runner services" which allows the user to focus on the logical aspects of the data processing implementation. Low level tasks like coordinating individual workers, sharding data sets etc. are managed by these runner services which allows for a level of abstraction. [3].

3.2 Components

There are four major components or parts to the Dataflow model:

1) Pipeline- A pipeline represents a group of computations wherein data is accepted as input, processed to provide insights and delivered as output. The input and output can be of the same type or different types [3].
2) PCollection- A PCollection is used to represent data in the pipeline. These classes act as containers to represent data that can be bounded or unbounded [3]. Thus this model can be used for batch as well as stream processing of data.
3) Transform-A transform is any data processing operation or any step in the pipeline. It takes a PCollection as input, performs a processing operation and produces an output PCollection[3].
4) I/O source and sink-These allow the pipeline to source data from multiple storages and formats. [3]

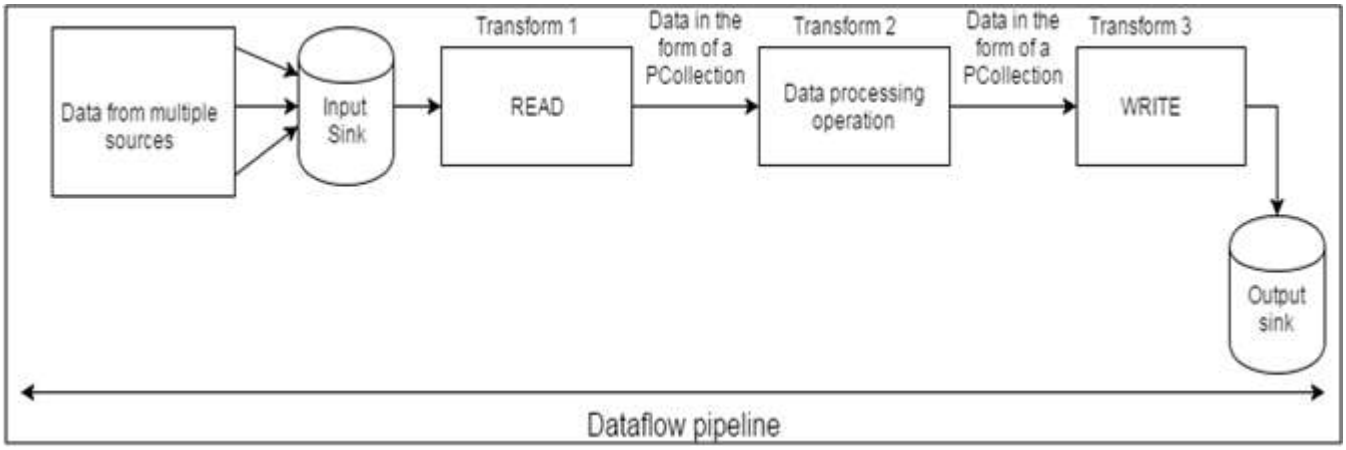

Figure 1: Dataflow Pipeline

4. Alternatives

Apache spark is one of the prime alternatives to Cloud Dataflow. Both these technologies use directed acyclic graph based processing. While Cloud Dataflow is supported by Google's underlying infrastructure, Spark is a standalone API and engine. This, however, means that while Dataflow allows for seamless integration with Google technologies, it is also bound to them. [4]

Cloud Dataflow is an excellent choice for companies who need production level processing support in the cloud whereas Spark is better suited for jobs where experimenting is more important. [4]

A possible mediator to this problem is Apache Beam. Using this, one could write their code and then decide which engine to use as this is compatible with different engines like Dataflow, Spark and Flink. [4]

5. Advantages

The advantages of Dataflow are:

- Seamless integration with existing Google technologies
- "Autoscaling" allows for flexible and automatic assigning of processing cores based on size of job[4]
- Faster than Hadoop when data is petabytes in size[2]
- Works well for production level jobs

6. Disadvantages

The advantages of Dataflow are:

- Bound to Google technologies
- Not suited for experimental data processing jobs

7. Acknowledgements

I am grateful to the teachers and experts who have guided me while writing this paper.

References

[1] F. Perry, Sneak peek: Google cloud dataflow, A cloud-native data processing service, Available at: https://cloudplatform.googleblog.com/2014/06/sneak-peek-google-cloud-dataflow-a-cloud-native-data-processing-service.html
[2] E. McNulty, Move over Mapreduce, Google's cloud dataflow has arrived, Available at: http://dataconomy.com/2014/06/move-map-reduce-googles-cloud-dataflow-arrived/
[3] https://cloud.google.com/dataflow/model/programming-model
[4] A.C. Oliver, Google cloud dataflow vs. Apache Spark:http://www.infoworld.com/article/3064728/analytics/google-cloud-dataflow-vs-apache-spark-benchmarks-are-in.html
[5] T. Akidau, T. Akidau, R. Bradshaw, C. Chambers, S. Chernyak, R. J. Fernandez-Moctezuma, R. Lax, S. McVeety, D. Mills, ´ F. Perry, E. Schmidt, et al. The dataflow model: a practical approach to balancing correctness, latency, and cost in massive-scale, unbounded, out-of-order data processing. PVLDB, 2015

Author Profile

Eashani Deorukhkar is an undergraduate student of Information Technology Engineering at Ramrao Adik Institute of Technology affiliated to Mumbai University. She is interested in the fields of Data Science and Data Analytics.