Apache Tez- Application Data processing Framework

Abhijt Thakre[1]

[1] *School of Informatics and Computing, Bloomington, IN 47408, U.S.A.*
[2] *Mechanical Engineer,Nagpur University, 2003*
[*] *Corresponding authors: abhijit.thakre@gmail.com*

project-000, April 11, 2017

There are lot of advancement in Hadoop framework from Hadoop1.0 to Hadoop 2.0. Hadoop 2.0 is layered architecture provides the YARN layer responsible for the resource management opens up the gate for developing different application engines on top of it. Apache Tez is one of such open source framework build on the top of YARN designed to build data-flow driven runtimes[1]. This paper focusses mostly on introduction to Apache Tez framework. It provides the insight of the architecture used in building Tez. It also tried to cover the technologies using apache Tez as unifying framework and performance improvement achieved due to that.

Keywords: Cloud, I524

https://github.com/cloudmesh/classes/blob/master/docs/source/format/report/report.pdf

This review document is provided for you to achieve your best. We have listed a number of obvious opportunities for improvement. When improving it, please keep this copy untouched and instead focus on improving report.tex. The review does not include all possible improvement suggestions and if you sea comment you may want to check if this comment applies elsewhere in the document.

Abstract : Don't use references in the abstract.

INTRODUCTION

In order to understand Tez as an framework we need to first dig into the history of Hadoop. Hadoop 1.0 have MapReduce as the central execution engine of its application. Any type of problem statement for analysis needs to be restructured to fit it to the map-reduce paradigm. It was also responsible for resource management and resource allocation. With Hadoop 2.0 these responsibilities got divided separately where YARN got the responsibility of general purpose resource management.

Apache Hive, Apache PIG , CASCADING which was initially using Hadoop1 now run on Hadoop2 on YarN. These listed data processing application including map reduce have certain set of requirement from the hadoop cluster in order to run efficiently. This is where Tez came in picture. Tez takes care of running the data processing application's efficiency and performance leaving the end user to only concentrate on the business logic.

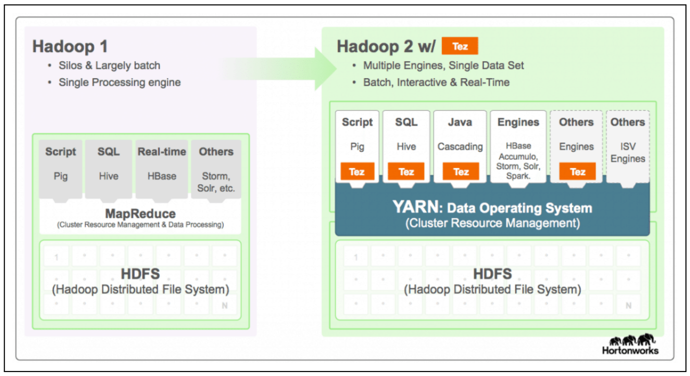

Hadoop 1
Silos & Largely batch
Single Processing engine
Script
Pig
SQL
Hive
Real-time
HBase
Others
Storm, Solr, etc.
MapReduce
(Cluster Resource Management & Data Processing)
HDFS
(Hadoop Distributed File System)
Hadoop 2 w/ Tez
Multiple Engines, Single Data Set
Batch, Interactive & Real-Time
Script
Pig
Tez
SQL
Hive
Tez
Java
Cascading
Tez
Engines
HBase Accumulo, Storm, Solr, Spark.
Others
Engines
Tez
Others
ISV Engines
YARN: Data Operating System
(Cluster Resource Management)
HDFS
(Hadoop Distributed File System)
Hortonworks

Fig. 1. [1]

TEZ TERMINOLOGY

Use bulletings or numbering for pointing out facts and try to keep it under 3 items or in this case you can use 4. But don't exceed that limit.

DAG – Direct Acyclic Graphs, it represent overall job. Vertex – Logical step in processing. It contains the details of user logic and dependent environment. Task – There can be multiple task in unit of work that vertex perform. Edge – This represents connection between producer and consumer vertices.

ARCHITECTURE AND IMPLEMENTATION

Tez was designed keeping in mind to address the problems which were not resolved by Hadoop. It was not build from the scratch but on the top of YARN layer to leavarage the advantages and work that were done for years on Hadoop. So Tez leavarage the discrete task based compute model,concept of data shuffling in map reduce,resource tentancy and multi-tenancy model and build in security from Hadoop .

Tez focuses mainly on below problem in addition.

- Without Tez, all the algorithm those needs to be executed on clusters needs to somehow translated to map-reduce api. This was impacting the efficiency and performance.However with Tez can naturally map the algorithm to execution engine in cluster.
- Tez provide additionally interface for various application and technology for data source and syncing.
- Performance.

TEZ-API

Tez provides below two API for defining the data processing.

Grammar

THis must be changed into This.

DAG API

THis API lets user define the structure for the computation. It lets user define the producer and consumers and how they talk to each other. This class of data processing application is represented as direct acyclic graphs

RUNTIME API

Using this API Layer Tez invokes the user code. This is where the actual code in the task to be executed is defined.

APPLICATION USING TEZ

Instead of below, just reference the subsections by using label and ref tags.

Below are the product those are updated to be used on TEZ framework to run on YARN.

Apache MapReduce

MapReduce is simple but powerful way of data processing. Tez product comes with build in map-reduce support.The configuration needs to be updated on YARN cluster for map reduce. Tez has inbuild map processor and reduce processor which provides the respective map reduce functionality.

Apache PIG

Unnecessary spacing between language and like word in the second sentence.

PigLatin is the scripting language provided by Apache PIG used to write complex ETL. Tez API can handle the multiple output which is usually the case for the procedural language like PIGlatin, helps in keeping the code clean and maintainable.

Apache Hive

Apache Hive is used to convert the query written in HiveQL to map reduce and execute on hadooop cluster.The HiveQL is translation into map reduce format is often inefficient. Using Tez the trees translate directly into DAGS.

Apache Spark

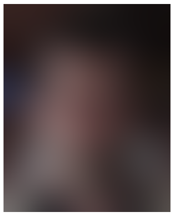
Aparche spark provides scaLA API for distributed data processing. The output of the spark is DAG of tasks that performance distributed computation. The end output of Apache spark i.e Spark DAG is successfully converted and executed using Tez API.

Overall Comment for improvement:
Content is not enough. Please add more information and describe them properly. And also use more references for each technology and important facts. The conclusion is missing. Please improve this to achieve your best in this paper.

REFERENCES

[1] "TEZ," Web Page, Indiana University, Mar. 2017. [Online]. Available: http://www.tez.apache.org

AUTHOR BIOGRAPHIES



Abhijit Thakre received his BE (Mechanical) in 2003 from The University of Nagpur.