

Big Data Platforms

Final Project

Apache Spark and Object Storage

Submission due to 09.02.2022

Background: Fault tolerance is critically important for persisting distributed datasets into physical storage. Scientists who wish to persist results of their analytic workloads are often not aware that results are distributed across the cluster and thus persisted in different parts where each task persist its own partition, without knowing if there are additional tasks that persist additional data partitions.

Problem statement: Fault tolerance algorithms, like FileOutputCommitter, are used both by Hadoop MapReduce and Apache Spark. There are many recent efforts to improve fault tolerance algorithms to interact efficiently with object storage, as many of them initially designed to work with file systems. Stocator and other different approaches are specifically designed to improve interaction of Hadoop MapReduce and Apache Spark with object storage.

Project definition: In this project students will explore and compare different approaches that enables Big Data engines an efficient way to persist distributed datasets into object storage (See [1],[2] references and find more sources). Students will also implement a prototype named “ExtendedObjectStorage” that will internally use MySQL database to support an atomic rename operation against object storage. ExtendedObjectStorage will expose an interface with “create_object, get_object, delete_object, create_directory, delete_directory, list_directory, rename_directory, rename_object”.

Project scope

Project will be submitted as a pdf paper accompanied with a prototype code
The following template should be used for the report

Template for the final report

Big Data Platforms

Small files and MapReduce

(Names)

(Submission date)

Abstract

Short description of the problem you are solving

Motivation and background

Describe what is MapReduce, object storage, how different object storage from HDFS. Explain differences of running MapReduce over data in HDFS vs data in the object storage. Explain what data partitions in the distributed data systems are. Explain why fault tolerance is important when persisting distributed data sets, how Apache Spark persist it's RDDs, what are FileOutputCommitter version 1 and version 2 about.

Stocator and object storage

Define the problem why Apache Spark when using FileOutputCommitter is non efficient when it persists RDDs in the object storage, comparing to using HDFS. Explain how Stocator works and how it achieves fault tolerance when persisting distributed datasets. Compare Stocator to other committers and emphasize pros and cons of each of the solutions.

Our approach

Define a framework over object storage that also contains a MySQL database that serves as a "catalog" of the data stored in object storage. This system suppose to provide atomic rename of the data persisted in object storage. Explain pros and cons of this system. Would you suggest cloud provider to use internally additional database to provide atomic rename?

Prototype

Write Python code with the prototype named "ExtendedObjectStorage" that will internally use MySQL database to support an atomic rename operation against object storage. ExtendedObjectStorage will expose an interface with "create_object, get_object, delete_object, create_directory, delete_directory, list_directory, rename_directory, rename_object".

Next steps

Suggest next steps to the solutions you proposed

Conclusion

Short conclusion of the work you did

Bibliography

List of all sources you were using in the project. At least 6 different sources

1. https://hadoop.apache.org/docs/r3.1.1/hadoop-aws/tools/hadoop-aws/committer_architecture.html
2. <https://arxiv.org/abs/1709.01812>

Points on grading

1. Extra points will be given to students who will create new GitHub project and upload their prototype there.
2. Extra points will be given to students who present deep solutions that address broad scope
3. The proposed solutions should address various aspects we learned during the course, like consistency, availability, fault tolerance, etc.
4. All the text students write should be their original and avoid as possible to copy text from other papers. In cases, when text need to be copied from another papers, make sure to reference what text is copied from other source and what is the source. Grade will be affected, If text is copied without proper reference.
5. Make sure you describe strong and weak points of your proposed solutions. Grade will be affected if there are additional weak or strong points than you managed to describe.