**15619 Project Phase 1 Report**

**Performance Data and Configurations**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Front end | Web service with HBase | Web service with MySQL |
| Query | q1 | q2 (small dataset) | q2 (small dataset) |
| Scoreboard request ID |  | 9022 |  |
| Instance type |  | m1.large |  |
| Number of instances |  | 5 |  |
| Queries Per Second (QPS) |  | 1072.4 |  |
| Error rate |  | 0.00 |  |
| Correctness |  | 91.00 |  |
| Cost per hour |  | 1.095 |  |

**Task 1: Front end**

**Questions**

1. Which front end system solution did you use? Explain why did you decide to use this solution.

2. Explain your choice of instance type and numbers for your front end system.

3. Did you do any special configurations on your front end system? Explain your design decisions and provide details here.

4. What is the cost to develop the front end system.

**Task 2: Back end (database)**

**Questions**

1. Describe your table design for both HBase and MySQL. Explain your design decision.

Hbase table has 4 columns in one family. HBASE\_ROW\_KEY(composed of user\_id and time from query), tweet\_id, sentiment, tweet. As the information to store has no complex relationships between them, only one table will suffice. The reason to combine user\_id and time to one column as row key is that Hbase has good performance for random read and will become slow if there is two many scan and compare operations.

2. What is the cost to develop your back end system.

The cost is to process the twitter JSON data to the tab separated file. It will run for linear time according to the input twitter data.

**Task 3: ETL**

Since ETL was performed for both HBase and MySQL, you will be required to submit information for each type of database.

MySQL:

1. The code for the ETL job
2. The programming model used for the ETL job and justification
3. The type of instances used and justification
4. The number of instances used and justification
5. The spot cost for all instances used
6. The execution time for the entire ETL process
7. The overall cost of the ETL process
8. The number of incomplete ETL runs before your final run
9. Discuss difficulties encountered
10. The size of the resulting database and reasoning
11. The time required to backup the database on S3
12. The size of S3 backup

HBase:

1. The code for the ETL job
2. The programming model used for the ETL job and justification
3. The type of instances used and justification
4. The number of instances used and justification
5. The spot cost for all instances used
6. The execution time for the entire ETL process
7. The overall cost of the ETL process
8. The number of incomplete ETL runs before your final run
9. Discuss difficulties encountered
10. The size of the resulting database and reasoning
11. The time required to backup the database on S3
12. The size of S3 backup

**Questions**

1. Describe a MySQL database and typical use cases.
2. Describe an HBase database and typical use cases.

Hbase is stored on slave nodes on AWS and we first import data to Hbase master cluster. Then we can import data to HDFS and then do map-reduce job on all nodes to store input data in Hbase. Each time we receive a query from front end, we have a GET query to Hbase and get the corresponding tweet information which is returned to front end.

1. What are the advantages and disadvantages of MySQL?
2. What are the advantages and disadvantages of HBase?

Hbase has good performance when doing random read operations. It is also faster when deploying data onto slave nodes. Hbase can also store bigger data than MySQL. However, Hbase is not tuned for all kinds of queries. We can only configure it to run certain kind of operations. Also, Hbase does not support JOIN operations.