Clustering Algorithm on CE operator

pidl etabyte-scale In-memory Database Lab.
정준영

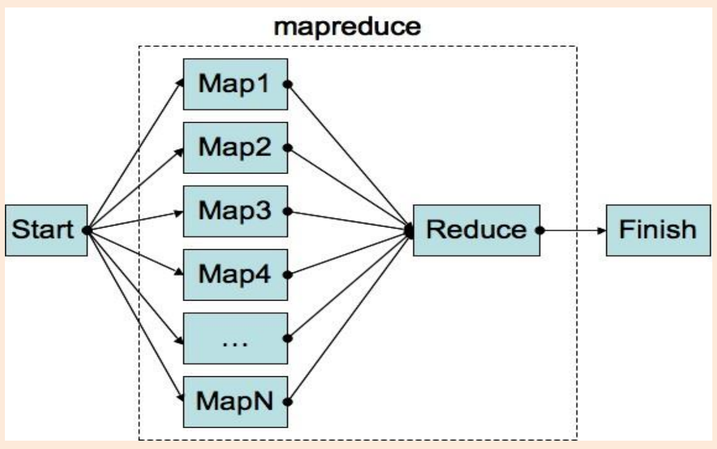
Goal

1. Implement clustering algorithm (K-Means) on CE, especially using Map Reduce
2. Compare the performance with Hadoop & SPARK

Introduction

Hadoop is an open-source software framework for storing and large scale processing of data-sets. It supports HDFS, which is a distributed and **DISK-based file-system** written in Java. With the HDFS, Hadoop, has powerful parallel data processing : **Map Reduce**

Map Reduce is the ability to divide dataset, and run it in parallel over multiple nodes.

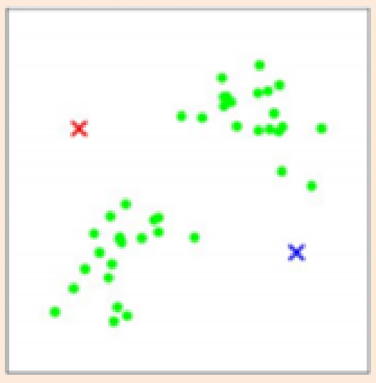


CE (Calculation Engine) provides interfaces which integrate separated many components HANA, commercial **in-memory** DBMS. **If Map Reduce technique can be implemented on in-memory DBMS with CE, it can show better performance than Hadoop's disk-based DBMS.**

SPARK is separated, fast, and Map-Reduce like engine. It has **in-memory** data storage for very fast iterative queries. CE and SPARK are both based on in-memory system, so comparison between CE and SPARK is also meaningful.

Implementation

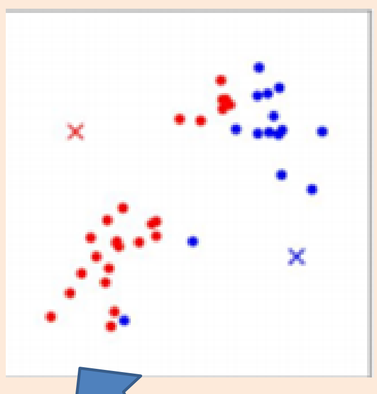
1. Idea of K-Means using Map Reduce



Initial Points and Center of cluster(Centroid)

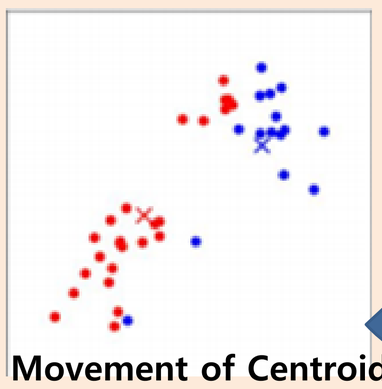
Point-Cluster Mapping

This stage is performed by 8 machines simultaneously.



Terminate Condition

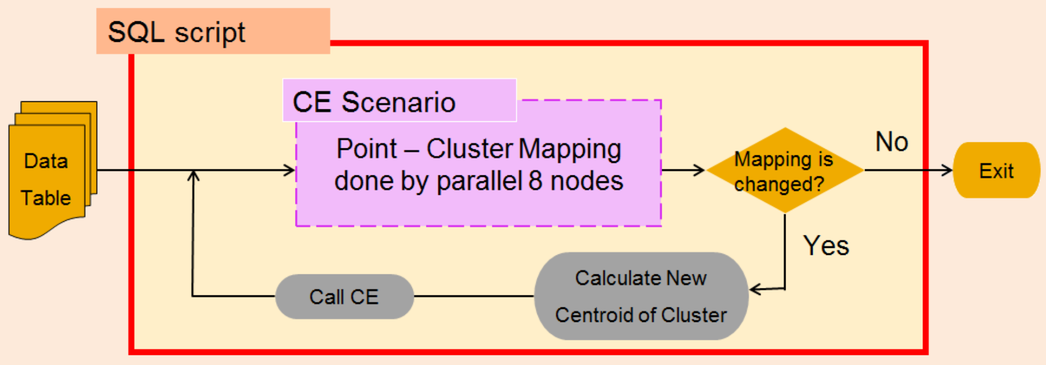
Go to previous stage until Point-Cluster mapping set is not changed



Movement of Centroids

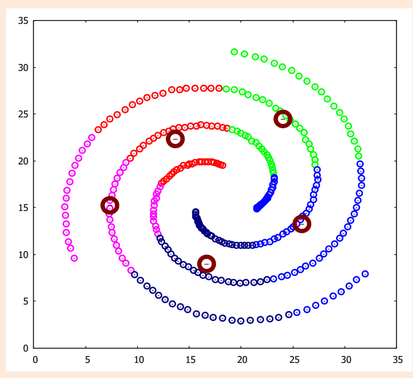
Coordinates of new centroids are average of mapped points' coordinates

2. Implemented on CE

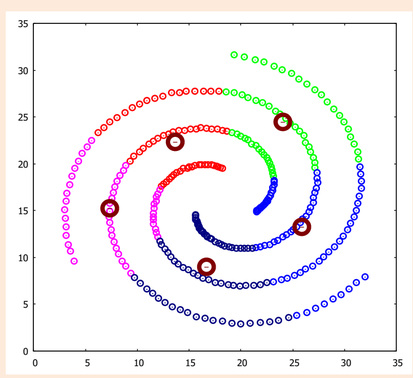


Result

1. K-Means of CE Validity Test



Hadoop's output



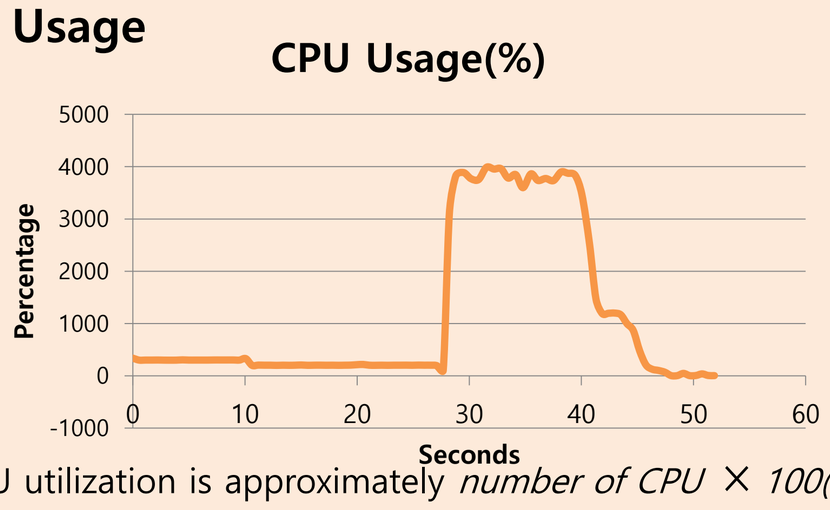
HANA's output

2. Test Environment

Number of nodes : 8
Number of CPUs per node : 40
Clock rate of CPU : 2.40GHz

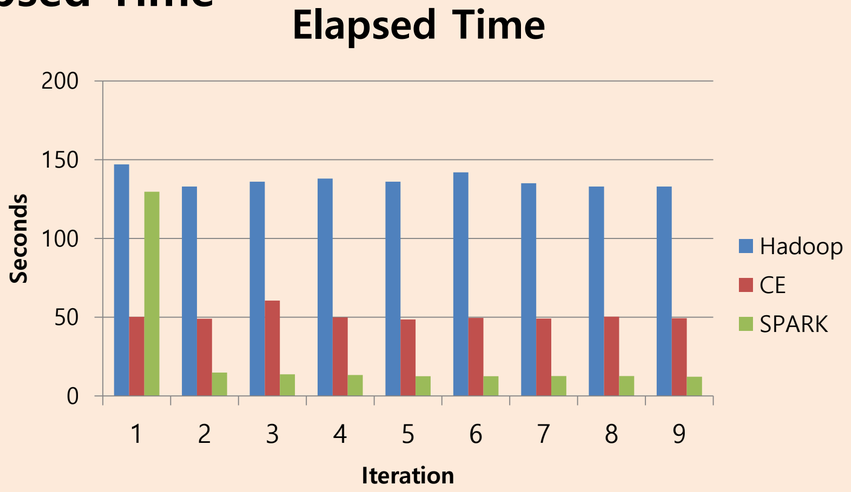
Test data set is derived from OpenStreetMap
It is consist of 1.9billion geographical points and its size is 77GB on memory

3. CPU Usage



CPU utilization is approximately *number of CPU × 100(%)*
Memory Usage is NOT change even if clustering is started

4. Elapsed Time



CE shows elapsed time which is ***reduced by 63% than Hadoop***

Conclusion

With high CPU usage and low memory usage, in-memory based DBMS performs about 3 times faster than Hadoop, which is disk-based file system. This results are easy to be expected, because disk-based file system takes long time to bring data from DISKs but, in-memory DBMS don't have to do that.

But, it shows lower performance than "SPARK". Several reasons of lower performance could be found by using profiling tool and they are same as below.

- Do merging only in one node
- Update some values to physical table
- Do full scanning whole table, when calculating new centroid coordinates

Future Work

SPARK has also flaws. First, it must take long time for 1st iteration. Because data must be loaded in memory from DISKs. Seconds, SPARK is operated on JVM which consume more memory than original data because of its overhead. In this experiment JVM consumes memory 50GB/node. But, CE does not have such problems. So, if a new design on CE can fix above 3 flaws in Conclusion section, CE performs better performance in Map Reduce technique.