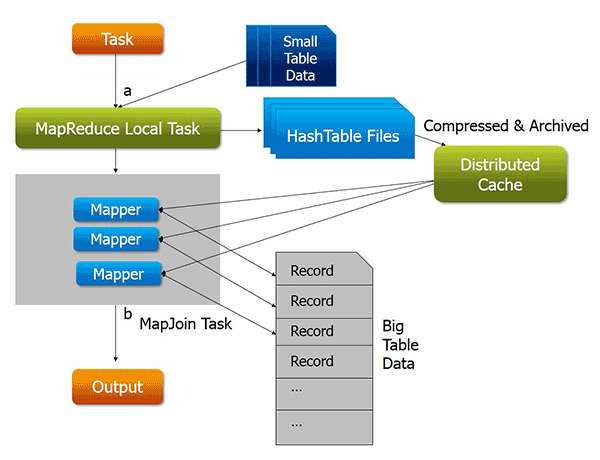
Explain the following in brief with an example.

● Map side Join

Here in the map joins we are storing the smaller files in the cache and when the original join Map Reduce task is running, it moves the data in the hash table file to the Hadoop distributed cache, which populates these files to each mapper’s local disk.



Map side join helps in reducing the time taken for shuffle and sort and as well as the reducing.

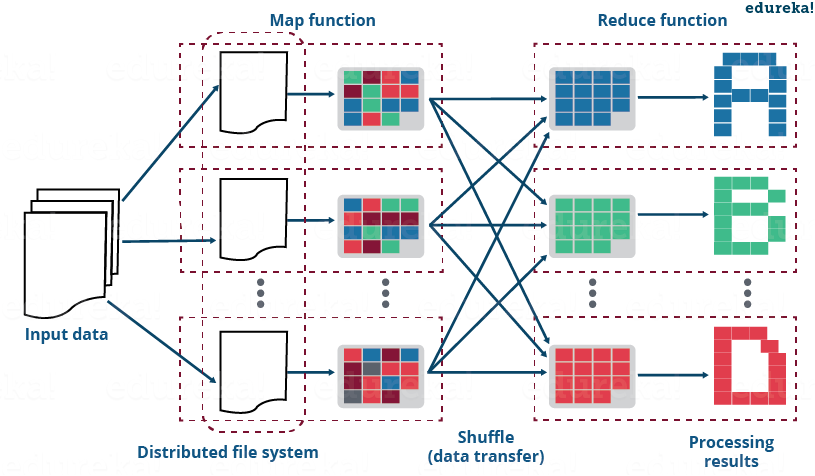
● Reduce side Join

Reduce side joins are simpler in functioning. Also the data should not to be necessarily structured. This type of the join is less efficient. Also used when both the data sets are very large and cannot be stored in the cache memory.

Also the time taken is more than that of the map side join as shuffle and sort and reduces phase’s takes place.

Here one has to use the multiple inputs in order to do reduce side join.

In the reduce side multiple inputs are taken into the account the passed to reducer the reducer combines the values from both the mapper for the same key. The structuring of the data has to done by the user itself. Hence much tedious process than that of the map side join.



● Bucket Map Join

In this type of the join the data is first divides in buckets. This division of the buckets is done according the hash value of the column so the rows stored in the buckets of the each of the database is same. Hence the joining process bucket by bucket is much faster than that of the normal join.

If the same join is done using the map side join along with the buckets is much more efficient hence called as the Bucket map join.

By default join in hive done using map side join only

● SMBM Join

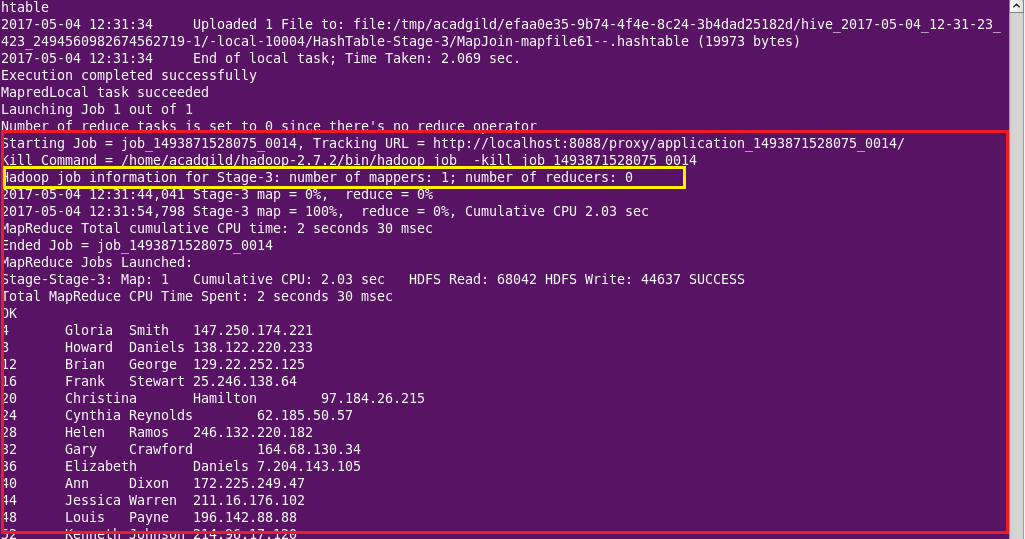
**Sort-Merge-Bucket map join**

**This type of similar to above bucket map join. But the table which was earlier just bucketed is also sorted and hence more optimization of the map side join will take place. As the processing is much easier on the sorted data than that of the unsorted data.**

**Examples**

Sort-Merge Bucket Map join

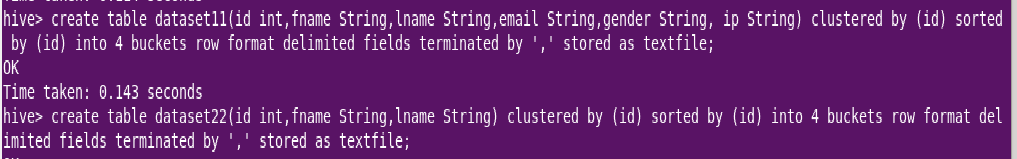


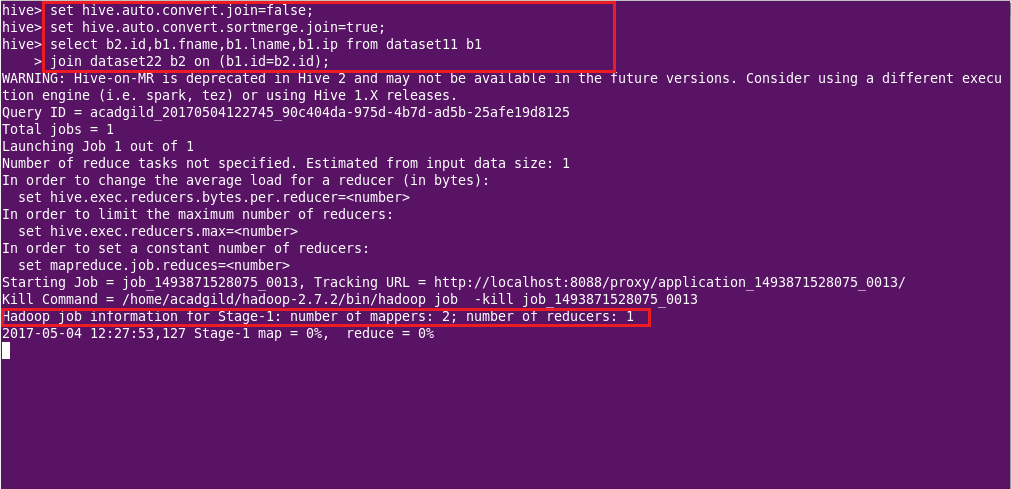


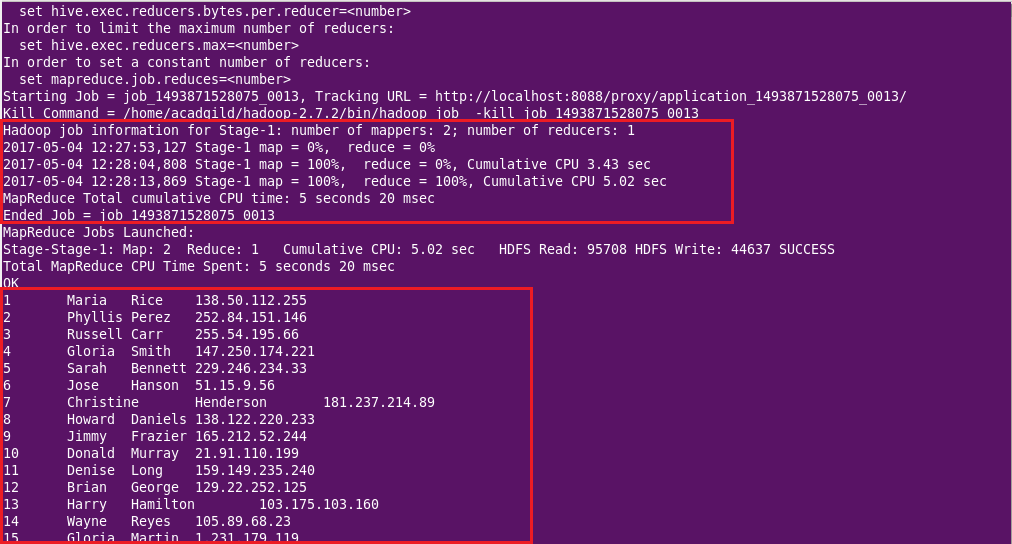
Here the auto convert join is set to true hence map side join takes place. Number of reducers are zero.

This is also the example of the map side as we can see that only map process is running. Number of reducers re zero.

Sort-Merge Bucket join





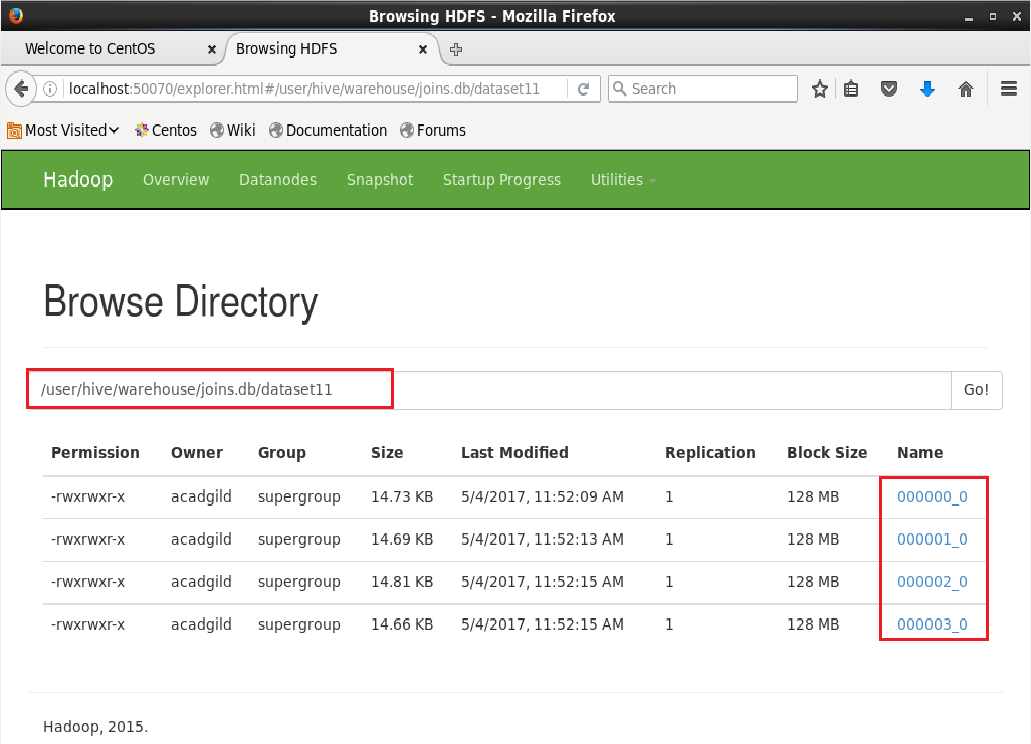


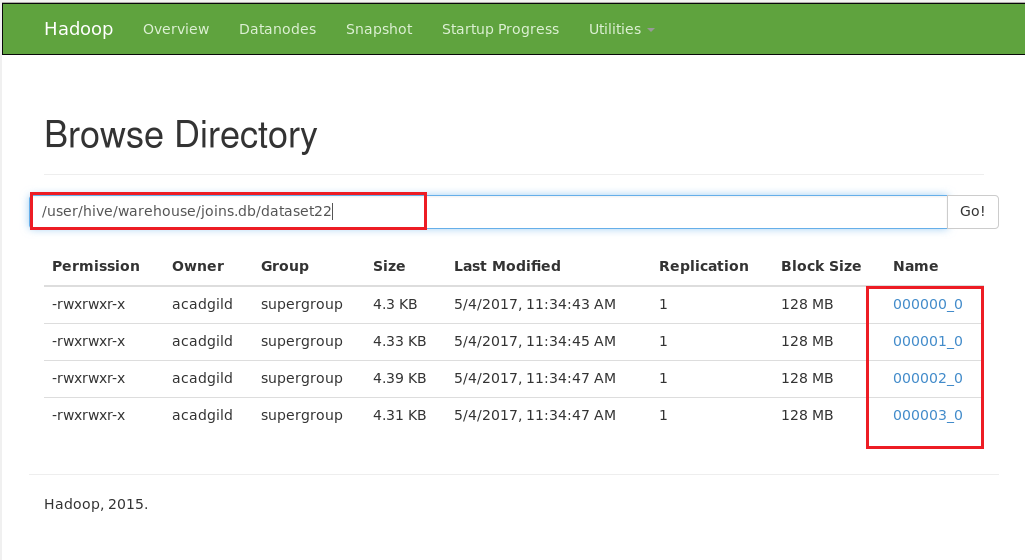
Here we can see that the join is done the bucketed tables but both map and reduce operations are being done.

This is possible by setting the auto.convert config of join to false. Here sort merge optimization of join is set to true.

Here we can see that number of reducers is one and hence it is a reduce side join.

Bucket Map join





Here we can see that the tables dataset11 and 22 are bucketed with 4 buckets

