1. *Map side Join*

If datasets are large, reduce side join takes place.

The I/P for to each map must be partitioned and sorted in a specific way. Each input dataset must be divided into the same number of partitions, and it must be sorted by the same key (the join key) in each source.

All the records for a particular key must reside in the same partition and which is mandatory. A map-side join can be used to join the outputs of several jobs that had the same number of reducers, the same keys and output files that are no bigger than the HDFS block size.

Ex: func ::= tbl(<class>,”<path>”);

*2) Reduce side Join*

In case one of the dataset is small, map side join take place.

In map side join, a local job runs to create hash-table from content of HDFS file and sends it to every node. Reduce-Side joins are simpler than Map-Side joins since the input datasets is not required to be structured. But it is less efficient as both datasets are

to go through the Map Reduce shuffle phase. The records with the same key are brought together in the reducer. We can also use the Secondary Sort technique

to control the order of the records.

SET hive.auto.convert.join=true;

3) Bucket Map Join

The data should be bucketed on the keys used in the ON clause and the number of buckets for one table must be a multiple of the number of buckets for the other table.

if these conditions met, Hive can join individual buckets between tables in the map phase, because it does not have to fetch the entire content of one table to match against each bucket in the other table.

Ex:

-> set hive.optimize.bucketmapjoin=true;

->SET hive.auto.convert.join=true;

4) SMBM Join

these joins can be converted to SMB map joins also.

these joins are used wherever the tables are sorted and bucketed.

The join boils down to just merging the already sorted tables, allowing this operation to be faster than an ordinary map-join.

Ex:

-> set hive.enforce.sortmergebucketmapjoin=false;

-> set hive.auto.convert.sortmerge.join=true;

->set hive.optimize.bucketmapjoin = true;

-> set hive.optimize.bucketmapjoin.sortedmerge = true;