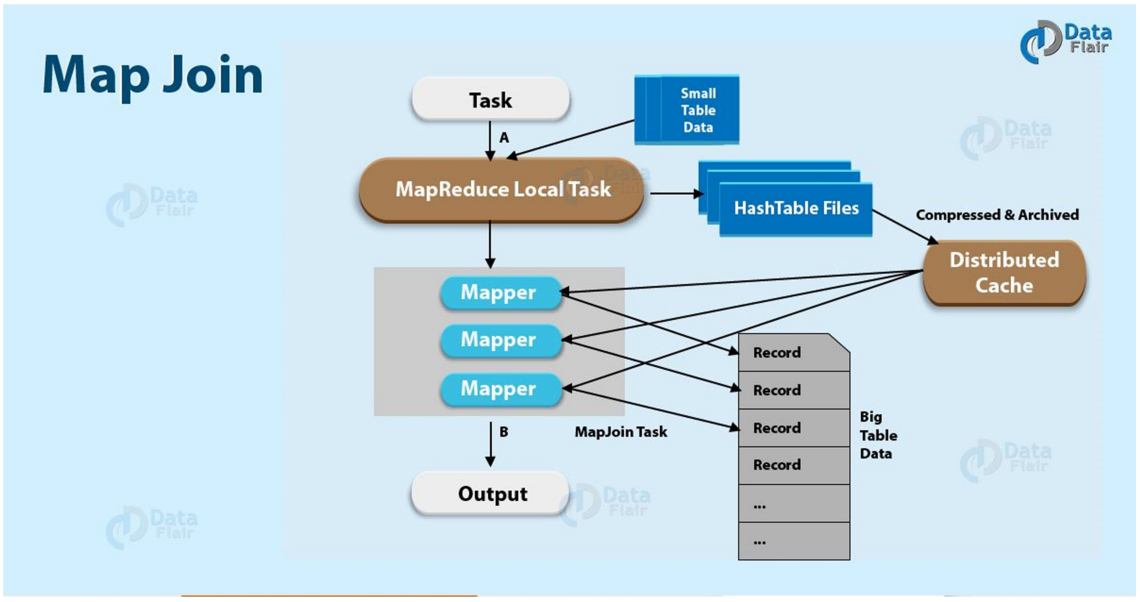
Joins in Hive

What is Map Join in Hive?

- If all but one table is small, the largest table can be streamed through the mappers while the small tables are cached in memory.

- Hive can do all the joining map-side, since it can look up every possible match against the small tables in memory, thereby eliminating the reduce step required in the more common join scenarios.

- Even on smaller data sets, this optimization is noticeably faster than the normal join. Not only does it eliminates reduce steps, it sometimes reduces the number of map steps, too.



- Before Hive v0.7, it was necessary to add a hint to the query to enable this optimization.

```
SELECT /*+ MAPJOIN(d) */
s.ymd, s.symbol, s.price_close, d.dividend
FROM stocks s JOIN dividends d
ON s.ymd = d.ymd AND s.symbol = d.symbol
WHERE s.symbol = 'AAPL';
```

- Running this query versus the original on a fast MacBook Pro laptop yielded times of approximately 23 seconds versus 33 seconds for the original unoptimized query, which is roughly 30% faster using our sample stock data.

- The hint still works, but it's now deprecated as of Hive v0.7. However, you still have to set a property, hive.auto.convert.join, to true before Hive will attempt the optimization. It's false by default:

```
hive> set hive.auto.convert.join=true;

SELECT s.ymd, s.symbol, s.price_close, d.dividend
FROM stocks s JOIN dividends d
ON s.ymd = d.ymd AND s.symbol = d.symbol
WHERE s.symbol = 'AAPL';
```

- Note that you can also configure the threshold size for table files considered small enough to use this optimization. Here is the default definition of the property (in bytes):

```
hive.mapjoin.smalltable.filesize = 25000000
```

Bucket Map Join

- This optimization can also be used for larger tables under certain conditions when the data for every table is bucketed.

- The data must be bucketed on the keys used in the ON clause (i.e join column) and the number of buckets for one table must be a multiple of the number of buckets for the other table.

- When these conditions are met, Hive can join individual buckets between tables in the map phase, because it does not need to fetch the entire contents of one table to match against each bucket in the other table.

- However, this optimization is not turned on by default. It must be enabled by setting the property hive.optimize.bucketmapjoin:

```
set hive.optimize.bucketmapjoin=true;
```

Sort Merge Join

- If the bucketed tables actually have the same number of buckets and the data is sorted by the join/bucket keys, then Hive can perform an even faster sort-merge join. Once again, properties must be set to enable the optimization:

```
set hive.input.format =
org.apache.hadoop.hive.ql.io.BucketizedHiveInputFormat;

set hive.optimize.bucketmapjoin = true;

set hive.optimize.bucketmapjoin.sortedmerge = true;
```