

Choosing Presto Join and Sort Algorithms

Join Algorithms


There are two types of join distributions:

- Partitioned: each node participating in the query builds a hash table from a fraction of the data
- Broadcast: each node participating in the query builds a hash table from all of the data (data is replicated to each node)

The default join algorithm used in Presto is distributed PARTITION join. This algorithm partitions both the left and right tables using the hash values of join keys. Partitioned join uses multiple worker nodes for processing queries; it is generally faster and takes less memory.

In some cases where one of the join tables is very small, the overhead of partitioning data through the network using distributed PARTITION join might exceed the benefit of broadcasting the whole table to all the nodes participating in the join operation. In those cases, 'BROADCAST' join might perform better. If you use the BROADCAST join, specify the large table first in the join clause. 'BROADCAST' join can be enabled specifying the following magic comment.

```
-- set session join_distribution_type = 'BROADCAST'
```

 This option will use more memory as the right join table will be copied to all nodes.

Sort Algorithms

For queries with ORDER BY clause, Presto uses Distributed sort by default; the sorting operation runs in parallel on multiple nodes, and a single node merges the final results. However, there might be cases where a single node sort operation performs better. For such cases, the distributed sort can be disabled by using the following magic comment.

```
-- set distributed_sort=false
```

 Disabling distributed sort will use more memory of a single node; the query might fail due to the maximum memory limit.