Amazon Redshift : Avoid data redistribution

Abhishek Tiwari

Citation: *A. Tiwari*, ”Amazon Redshift : Avoid data redistribution”, Abhishek Tiwari, 2016. doi:10.59350/pj1vz-fz749

Published on: March 14, 2016

When using Amazon Redshift, distribution style plays an important role in optimising the table design for best performance. In a nutshell, table's distribution style dictates how the data is distributed across Redshift node and slices. A key objective is to avoid the data redistribution during query execution or runtime. This is accomplished by locating or co-locating the data where it needs to be before the query is executed. For instance, if a query is performing join over two tables, to avoid the redistribution of data, data from two tables can be co-located by planning an appropriate distribution style.

Cost of data redistribution

Amazon Redshift query execution engine ships with an MPP-aware query optimizer. Redshift's query optimizer determines where the block of data need to reside to execute the most optimized query. This means Redshift query execution engine may need to move or redistribute data from one node or slice to another physically during the runtime. This can happen for two reasons - first when performing joins or aggregates and second when trying to distribute the workload uniformly among the nodes in the cluster. The cost of data redistribution can be substantial, and often it will slow down query performance. In particular, moving data from one node will have a major impact on network traffic.

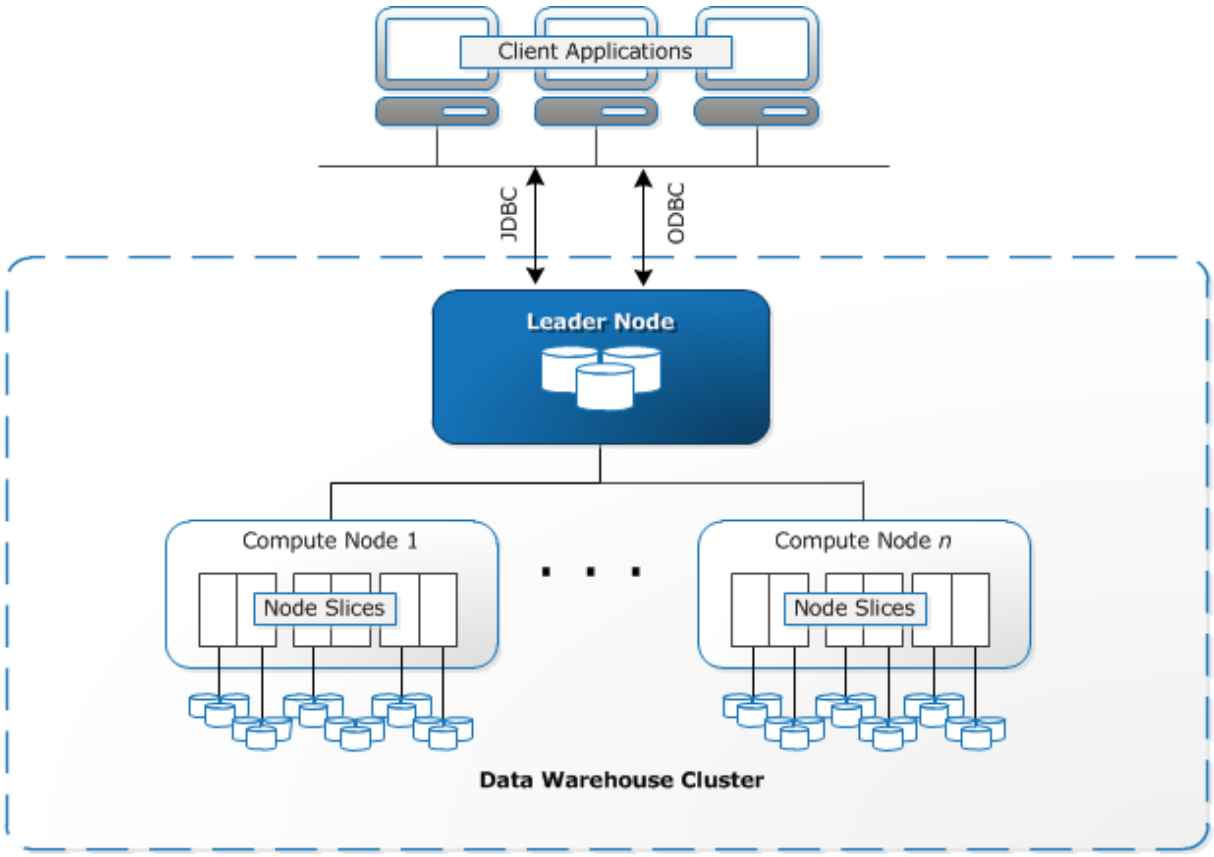


Figure 1: Amazon Redshift data warehouse architecture: Nodes and Slices. Leader node manages communications and query execution plan. Compute nodes execute the compiled queries. A compute node is partitioned into slices.

Table distribution styles

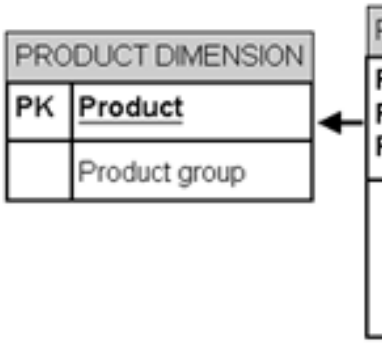
Amazon Redshift supports three different types of table distribution styles: Even, Key and All. Even distribution is the default distribution style for Redshift. Please note these distribution styles are applied at table level but the choice of distribution style often depends on the type of schema used in your database design. If you are using a star schema, a variant of star schema or a totally denormalised schema - you have to factor these in your table distribution style decision.

- Even distribution - data is distributed across the slices in a round-robin fashion. This is ideally used when a table does not participate in the join.
- Key distribution - data is distributed according to the values in one column. If two tables distributed on the joining key, data is co-located on the slices according to the values in the joining columns.
- All distribution - A copy of the entire table is distributed to every node.

Denormalized schema

If a table is largely denormalized and does not participate in joins as a rule of thumb always use the Even distribution.

Common columns



Distribute the fact table and its largest dimension table on their common columns.