International Journal of Computer Trends and Technology
ISSN: 2231 – 2803 / https://doi.org/10.14445/22312803/IJCTT-V71I2P107

Volume 71 Issue 2, 40-44, February 2023
© 2023 Seventh Sense Research Group®

Original Article

Amazon Redshift: Performance Tuning and Optimization

Amol Kulkarni

SAP Solution Architect, Data-Core Systems Inc, Bristol PA 19007

Received: 05 January 2023 Revised: 07 February 2023 Accepted: 17 February 2023 Published: 28 February 2023

Abstract - *Data is the lifeline for any enterprise in today's world. Exabytes of data are generated daily due to billions of transactions worldwide, social media, smartphones, and gaming apps use. It is most crucial for an enterprise not only to collect exabytes of data in the OLTP ERP System but also to analyze it for Decision making, predictive analysis, trend analysis and gain insights into business using the Data Warehouse OLAP System. With the advent of Cloud technology, it is possible to handle exabytes of data in cloud-based OLTP ERP systems and analyze it in cloud-based Data Warehouse OLAP Systems. There are various cloud-based data warehouse systems available in the market, but an enterprise has to choose the right system based on cost, integration with the OLTP ERP System, integration with existing business intelligence tools, security, scalability, data loading and query performance. This paper demonstrates how the Amazon Redshift performance optimization options made it the first choice for any enterprise to load huge amounts of data and provide it quickly for decision-making and analytic reporting purposes.*

Keywords *- Amazon S3, AWS, Business Intelligence (BI), Cloud, Data warehouse, ETL, Machine Learning (ML), SQL.*

1. Introduction of Amazon Redshift

Amazon Redshift is a fully managed petabyte-scale AWS cloud-based data warehouse service introduced in 2013 by Amazon Web Services AWS. It is based on columnar storage, massively parallel processing MPP and distributed keys functionality for faster performance against a high volume of data. It allows you to analyze your large volume of data using different BI tools. Amazon redshift is 5 times less costly and 3 times faster than traditional data warehouse and tightly integrated with Amazon S3. All the recommendations for performance improvement in Amazon Redshift are automated using Machine Learning features.

2. Literature Review

Different cloud-based data warehouses like Amazon Redshift, Snowflake, Google Big Query, and Azure Databricks are available in the market, mainly used for collecting enormous data and data analytics and reporting. Choosing the right data warehouse is based on different factors like tight integration with existing OLTP ERP Systems, easy scaling for collecting a large volume of data, high performance while running the queries on large datasets, lower cost, ease of maintenance, robustness and security, expertise to implement etc. Amazon Redshift is a top cloud-based data warehouse on the list. If the organization is running on AWS Cloud as an OLTP ERP system, then Amazon Redshift is the first choice due to its tight integration with AWS apart from all the above factors.

3. Architecture of Amazon Redshift

Amazon Redshift Architecture comprises Client Applications, Connections, Clusters, Leader Nodes, Compute Nodes, Node Slices, Data Storage and Databases.

3.1.1. Client Applications

Amazon Redshift Integrates with Business Intelligence and Analytical tools like Microsoft Power BI, Tableau, Amazon Quicksight, Microsoft SQL Server Reporting Services, SSRS etc.

Different SQL Clients integrate easily with Amazon Redshift as it is based on Open Standard PostgreSQL. PostgreSQL supports both SQL and JSON querying and performs complex queries well against large datasets.

3.1.2. Connections

In order to work with data in Amazon Redshift Cluster, you have to connect the client applications using JDBC, Python and ODBC Drivers. JDBC monitor the connections of Amazon Redshift with client applications, and ODBC allows client applications to interact with the live data of Amazon Redshift. These drivers support IAM users, Single Sign On SSO, Authentication and high performance for Amazon Redshift Enhancements.

3.1.3. Cluster

Amazon Redshift's main component consists of Leader Nodes and Compute Nodes. Leader Nodes handle communication with client applications, and it also interacts with compute nodes. Each Cluster runs on Amazon Redshift Engine and contains one or more databases. After creating the Cluster, we can perform operations like resizing the Cluster, pausing and stopping Cluster, deleting and shutting down the Cluster using Amazon Redshift Console or AWS CLI and Amazon Redshift API.

3.1.4. Leader Nodes

It is the main node which receives queries from client applications, analyze them into different parts, prepares an execution plan, compiles and distributes the code and distributes the data to each compute node. It coordinates for parallel execution of the query with compute nodes, aggregates the results from the compute nodes, and finally returns the query results to client applications. Some SQL Functions like create_schema are performed only on Leader Node and not Compute Nodes.

3.1.5. Compute Nodes

Compute Nodes receive the compiled code from the leader node and execute it, and send the results back to the Leader note for final aggregation.

Compute Nodes of three types: 1.RA3 is for high-speed caching and high bandwidth networking. It charged as per the space used and hence cheaper. 2. Dense Compute Node DC2 is used for high computation power with less data in GB. 3. Dense Storage Node DS2 is used for a high volume of data.

Compute nodes have their own disk storage based on the above 3 node types, their own CPU and Memory.

3.1.6. Node Slices

Compute Nodes are divided into different slices. Each slice has its memory and disk space. Each node slice performs the operation of the task to which it is assigned, and each slice works in parallel to complete the operation of the task. There are 2 to 16 slices per compute node, depending on the node types.

3.1.7. Redshift Manage Storage RMS

Data is stored in RMS. It allows to scale of the Data Warehouse to a petabyte and pays for each Cluster's computing and storage capacity in the Redshift Data Warehouse. RMS makes the Redshift 99.99% available all through the year.

3.1.8 Databases

Each Redshift Cluster is based on one or more databases, and data is stored in each Compute Node. Amazon Redshift is an RDBMS based on Postgre SQL designed to provide high performance and reports on very large datasets.

3.2. Data Loading in Amazon Redshift

We can load data to redshift using the most popular COPY command, which supports different file formats like CSV, JSON, and AVRO and is mainly used for loading from Amazon S3, Dynamo DB, and EMR systems and is used for bulk loads. COPY command compresses the data in the Redshift tables, loads data in Parallel, and performs simple conversions. Other redshift data loading options are AWS Data Pipelines and the Insert INTO command. We can also use data to redshift from amazon S3 using dense copy, which recreates and repopulate the tables using bulk insert.

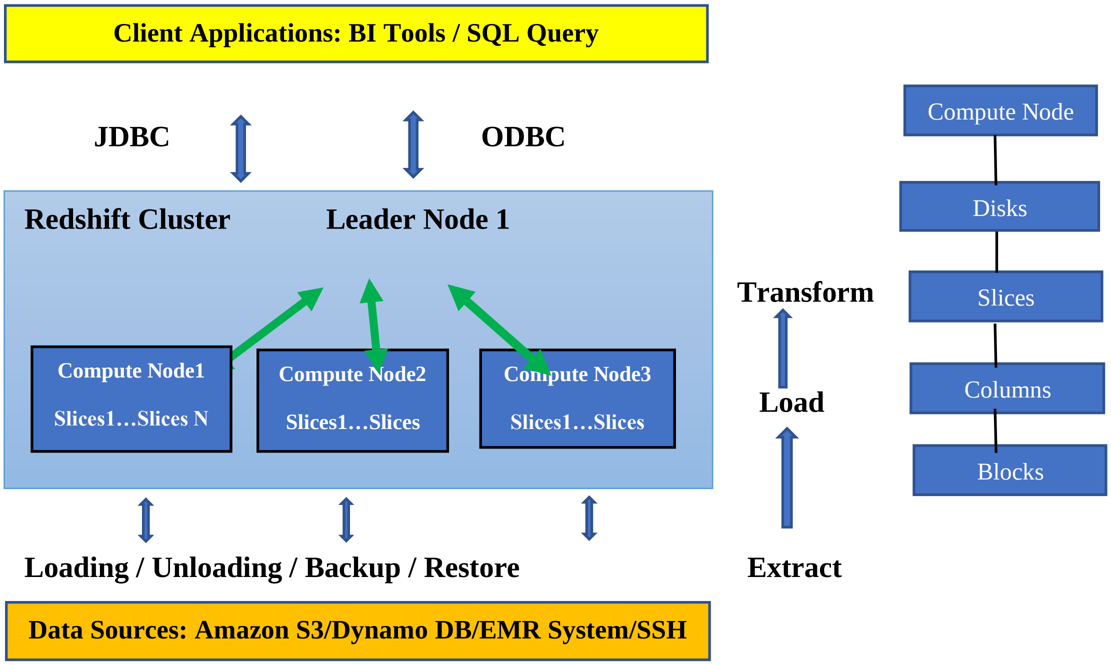

Fig. 1

4. Performance Tuning and Optimization in Amazon Redshift

Amazon Redshift has some box performance optimization features for handling huge data loads and high-speed query execution against enormous volumes of data. Compressed, sorted and distributed key table structures provide high query performance. We are going to discuss different techniques of performance optimization of Amazon Redshift below,

4.1. Compression

When data is loaded in Amazon Redshift tables, it is compressed 3 to 4 times. This reduces the storage footprint, reduces the I/O and improves the performance of the queries. Using advisor, we can get the list of tables which are uncompressed and suggestions to compress them. When we load data using copy command, data is compressed in the table, and each column is applied with the default encoding type, which helps save the data footprint and improves the performance of the query. Default Compression Type AZ64 is applied for column types like INT, BIGINT, TIMESTAMP etc. and default encoding type LZO is applied for VARCHAR and CHARACTER columns.

Select "column", type, encoding from PG_TABLE_DEF
where tablename = 'sales'

4.2. Distribution Style

It is a table property which explains how the data is distributed among different compute nodes and their slices. It helps in massively parallel processing for query performance improvement. The automatic table optimization property chooses the relevant distribution style among KEY, ALL, EVEN, and AUTO.

KEY: The values of the columns are hashed, and matching common columns are stored in the same slice. This is recommended for the high volume of tables. A particular Key goes to a particular slice only.

ALL: It distributes entire table data on each compute node and is hence used for a small table. Once the table grows, it automatically changes to KEY, EVEN.

EVEN: It is equal distribution of records to all slices. When Key distribution cannot produce even distribution, then EVEN is used.

AUTO: By default, all tables are set to AUTO distribution style.

If you use the join column between 2 tables as the distribution key and sort key, it improves the query's performance a lot.

Select "column", type, distkey from PG_TABLE_DEF where tablename = 'sales'

4.3. Sort Keys

Frequently used columns are sorted to reduce I/O operations and optimize the performance of the query. When a particular sorted column is used in the query, it picks up only the range of the values of that column for which the query is applied. For example, if the TIMESTAMP column is sorted, then only the range of data required for the given TIMESTAMP range is picked for the output instead of all data. This way, the I/O operations are reduced, and the query performance is improved.

Select "column", type, **sortkey** from PG_TABLE_DEF
where tablename = 'sales'

4.4. Redshift Advisor

It recommends the best performance options and cost-saving approaches based on the data inflow and outflow of the redshift cluster.

The Redshift advisor recommends the COPY command as the best data-loading approach to compress the data in the target redshift table. It suggests the best approach for deleting (vacuuming) the cluster data, which is not used to reduce the cost, and it suggests the optimization of manual workload management WML. All these recommendations can be applied by using an advisor and continuously recommended by analyzing the cluster data. The automatic table optimization feature recommends the sorting and distributing keys of the tables by analyzing the query patterns using Machine Learning ML. All the recommendations by the advisor and the automatic table optimization feature are applied during the time when the system is in minimum use, thereby making redshift a fully self-tuning system.

4.5. Analyzing and Improving Queries

Query Performance tuning and optimization is automated by redshift by means of automatic sort and distribution key applying on tables, automatic table optimizations, and automatic deleting (Vacuuming) the unused tables data. It analyzes the query pattern and provides the query execution plan insights like type joins of the table, cost of each step in the query and performance. These features are applied by means of Machine Learning which analyzes the workloads in the system. Different factors can affect the performance of the queries, like Hash Joins, Nested Loops, insufficient memory allocation to the query, Table statistics out of date etc. There are different ways to tackle these, like using merge joins in the query for hash join replacement, removing cross joins for nested loops, using the WLM tool for memory allocation, running analysis a plan for table statistics etc.

4.6. Workload Management WLM

It allows setting query priorities in the Cluster based on business needs. The sequence of query execution is based on memory allotment, query groups, query priorities and query monitoring rules. For example, short-running queries can be set to run first, and long-running queries can be run afterwards. Once the WLM is configured, we can allocate the memory and concurrency values of the workloads either automatically or manually. This way, the workloads on the Cluster are managed by means of WLM.

4.7. Concurrency Scaling

This feature allows redshift to support concurrent users and concurrent queries with high performance by adding additional clusters temporarily. Once currency queries are completed, the added Cluster is closed. Workload Management WLM enables Concurrency Scaling, and with the help of WLM, the short-running queries are allowed to be executed first, and long-running is executed later. Adding additional clusters for concurrency scaling is charged only for the time it is used and not after the query execution is completed.

4.8. Materialized Views MV

In redshift processing, the complex queries on large database tables are very expensive in terms of the system resources it occupies and the time it takes to perform the operations. Materialized views store the result of these complex queries in the form of a table; this way, when this query is executed, it provides the output very fast. Redshift automates the maintenance and use of Materialized views MV in different ways. MV provides the output incrementally whenever there is a change in the data in the table. Redshift also decides which MVs are frequently used and which are outdated and, based on that, finalizes the priority of which MVs to be maintained in the background. The goal here is to maximize the performance benefits of the MVs.

4.9. Redshift Spectrum

Redshift Spectrum offers the feature of querying the external data of Amazon S3 without actually loading it to Amazon Redshift tables. The Redshift queries, which need data from large datasets of Amazon S3, are run with the help of massively parallel processing of Redshift Spectrum.

5. Conclusion

Amazon Redshift is consistently innovating new things like AQUA, Data Sharing, Spectrum and different performance optimization features to make it an industry leader. It automates table optimization, Workload management using Machine Learning. Cost is a big factor when we compare amazon redshift with Snowflake, Google Big Query, or Microsoft Azure. Redshift's tight integration with Amazon S3 makes it the first choice for customers to use Amazon S3 as their ERP OLTP system. Easy deployment, maintenance, scalability, encrypted data while copying and best performance make it the top choice over any other data warehouses available in the market.

References

[1] Amazon Redshift website: Amazon Redshift System Overview : [Online]. Available: https://docs.aws.amazon.com/redshift/latest/dg/c_redshift_system_overview.html

[2] Amazon Redshift website: Amazon Redshift High Performance Queries:[Online].Available:https://docs.aws.amazon.com/redshift/latest/dg/c_challenges_achieving_high_performance_queries.html

[3] Amazon Redshift website: Amazon Redshift Automatic Table Optimization: [Online]. Available: https://aws.amazon.com/blogs/big-data/optimizing-tables-in-amazon-redshift-using-automatic-table-optimization/

[4] Amazon Redshift website: Amazon Redshift Automatic Workload Management WLM: [Online]. Available:https://docs.aws.amazon.com/redshift/latest/dg/automatic-wlm.html

[5] Amazon Redshift website: Amazon Redshift Concurrency Scaling: [Online]. Available: https://docs.aws.amazon.com/redshift/latest/dg/concurrency-scaling.html

[6] Amazon Redshift website: Amazon Redshift Implementing Workload Management: [Online]. Available: https://docs.aws.amazon.com/redshift/latest/dg/cm-c-implementing-workload-management.html

[7] Amazon Redshift website: Amazon Redshift Materialized Views: [Online]. Available: https://docs.aws.amazon.com/redshift/latest/dg/materialized-view-overview.html

[8] Amazon Redshift website: Amazon Redshift Database: [Online]. Available: https://docs.aws.amazon.com/redshift/latest/dg/r_Database_objects.html

[9] Amazon Redshift website: Amazon Redshift DynamoDB: [Online]. Available: https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/RedshiftforDynamoDB.html

[10] Anurag Gupta et al., “Amazon Redshift and the Case for Simpler Data Warehouses,” *ACM SIGMOD International Conference on Management of Data*, pp. 1917–1923, 2015. *Crossref*, https://doi.org/10.1145/2723372.2742795

[11] Forbes Council Post : Five Critical Reasons to Move Your Legacy Data Warehouse to the Cloud [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2020/02/26/five-critical-reasons-to-move-your-legacy-data-warehouse-to-the-cloud/?sh=5c866fb3205a

[12] Daniel Abadi, Peter Boncz, and Stavros Harizopoulos, “The Design and Implementation of Modern Column-Oriented Database Systems,” *Foundations and Trends in Databases*, vol. 5, no. 3, pp. 197–280 2012. *Crossref*, https://doi.org/10.1561/1900000024

[13] Caius Brindescu et al., “Integrate Etleap with Amazon Redshift Streaming Ingestion (preview) to make data available in seconds,” *AWS Big Data Blog*, 2021. [Online]. Available: https://blog.etleap.com/integrate-etleap-with-amazon-redshift-streaming-ingestion-preview-to-make-data-available-in-seconds

[14] Gerardus Blokdyk, *Amazon Redshift Complete Self-Assessment Guide*, Createspace Independent Publishing, 2017.

[15] Stefan Bauer, *Getting Started with Amazon Redshift*, Packt Publishing, 2013.

[16] Shruti Worlikar et al., *Amazon Redshift Cookbook: Recipes for building modern data warehousing solutions*, Packt Publishing, 2021.

[17] Simplilearn, Cloud Computing Architecture, 2019. [Online]. Available: https://www.simplilearn.com/cloud-computing-architecture-article

[18] Nandula Rohan Kausik, Nandula Nikhil Kartik, and G. Soma Sekhar, "Personal Expense Tracker utilizing Amazon Web Services," *International Journal of Computer Trends and Technology*, vol. 70, no. 10, pp. 8-14, 2022. *Crossref*, https://doi.org/10.14445/22312803/IJCTT-V70I11P102

[19] Chandan Patra, "Amazon DynamoDB: What It Is and 10 Things You Should Know," *Cloud Academy Blog*, 2022. [Online]. Available: https://cloudacademy.com/blog/amazon-dynamodb-ten-things/

[20] *AWS Redshift Architecture: 7 Important Components*, HEVO, 2017. [Online]. Available: https://hevodata.com/blog/redshift-architecture/

[21] Nay Lin Soe, "Concurrency Scaling in AWS Redshift," *Servian*, 2019. [Online]. Available: https://servian.dev/concurrency-scaling-in-amazon-redshift-f8a3a06cc087

[22] Mark Smallcombe, "We Tested Amazon Redshift Concurrency Scaling: Here are our Results," *Integrate*, 2021. [Online]. Available: https://www.integrate.io/blog/we-tested-amazon-redshift-concurrency-scaling-here-are-our-results/

[23] AWS Redshift Spectrum? [Online]. Available: https://ahana.io/answers/what-is-aws-redshift-spectrum/

[24] A Brief Summary of Compound Sort Key and Interleaved Sort Key with Performance Tests | Redshift, [Online]. Available: https://dev.classmethod.jp/articles/redshift-sortkey-usecase-en/

[25] Adam Gatt, Automate Your Amazon Redshift Performance Tuning With Automatic Table Optimization, 2021. [Online]. Available: https://noise.getoto.net/2021/10/06/automate-your-amazon-redshift-performance-tuning-with-automatic-table-optimization/

[26] AWS Snowflake vs. Redshift: Choosing a Modern Data Warehouse, [Online]. Available: https://www.stitchdata.com/resources/snowflake-vs-redshift/

[27] Akshay kumar, Dr.Divakar H R, and Dr.B R Prakash, "Elastic Search Usage of Analysis for Resource Scalability in AWS," *International Journal of Computer Trends and Technology*, vol. 69, no. 8, pp. 11-14, 2021. *Crossref*, https://doi.org/10.14445/22312803/IJCTT-V69I8P103