UNITED TECHNO
united we solve

www.unitedtechno.com

SNOWFLAKE PERFORMANCE TUNING:

Guide to 5 Best Practices

How can SQL queries be fine-tuned on a database without indexes? The task comes with challenges, but various techniques can be applied to accomplish the task.

Table of Contents

UNITED TECHNO
united we solve

The Need for Improved Data Warehousing

Businesses are focused on upscaling their data collection mechanisms that have made ‘Big Data’ a very popular term. Data sources are varied, including business transactions, social media, smart devices, and more. Data warehousing solutions have become the need of the hour. However, with cloud computing gaining significant traction, firms can choose a Software-as-a-Service (SaaS) for their data warehousing requirements.

With its features and flexible architecture, Snowflake has become the leader in the Big Data market. Multiple advanced query-tuning techniques are available on the platform, but with Snowflake indexes not being supported on default tables, performance tuning seems tough. In this e-book, we’ll explore the top 5 methods.

The warehousing market for unstructured data, including data not associated with any recognizable model, is estimated to grow by over 10% CAGR from 2019 through 2025 as enterprises leverage unstructured data for advanced analytics.

UNITED TECHNO

united we solve

Key Features of Snowflake

The features that make Snowflake a popular choice are:

1

SQL Support

Being a SQL-based platform, all standard and extended SQL commands are supported on Snowflake.

2

Graphical User Interface

The web-based interactive dashboard connects with the cloud seamlessly, making it convenient to monitor system usage and query data.

3

Command-Line Client

A separate Python-based tool can be downloaded for querying data and other functions.

4

Expansive Integration Capabilities

Snowflake can be integrated with multiple third-party tools, e.g. Google Cloud.

The Snowflake Architecture

Snowflake's architecture comprises not a single machine but three interconnected computer systems, each having separate hardware and software with auto-scaling capabilities. The layers, as illustrated in the diagram, include:

The Cloud Services Layer

Connections are framed on this layer, and Snowflake's query optimizer tunes the query by rewriting the code to maximize SQL query performance.

The Compute Services Layer

Queries are executed on a virtual warehouse that consists of a cluster of machines, and each node in the cluster is a computer with 8 CPUs, memory, and solid state disk (SSD) for temporary storage. The virtual warehouse storage, Local Storage, is significantly faster than disk storage.

The Cloud Storage Layer

Data is physically stored in blob storage. Held on a hard disk, cloud storage is referred to as *Remote Storage* in Snowflake.

Data warehouses are available in multiple shapes and forms. Snowflake is the market leader in data warehousing technology, holding a 19.24% market share.

Significance of Snowflake Performance Tuning

The licenses and hardware of the on-premise databases have to be purchased upfront, whereas Snowflake charges customers on a per-second basis for compute resource usage. For the performance tuning initiative to be a success, the customer has to set the priorities:

Lower Query Elapsed Time

Besides being a common priority for end-user queries, fast data ingestion and transformation ensure faster delivery of data for analysis.

Lower Costs

If the priority is to reduce the costs incurred, a more strategic approach needs to be in place to monitor the warehouse size and deployment.

Top Snowflake Performance Tuning Techniques

SEGMENTATION OF QUERY WORKLOADS

To maximize throughput and minimize latency, query workload segmentation is the most crucial method. Being a cloud-based platform, Snowflake supports unlimited virtual warehouses with shared access to a common data store. The EPP (Elastic Parallel Processing) architecture enables complex data science operations, ELT loading, and BI queries for the same data without any resource contention.

The common tendency is to separate workloads on the basis of department or team, but it's an accepted practice to separate workloads by their type rather than based on the user group/s. Business intelligence queries for marketing users can be hosted in one warehouse, while a separate virtual warehouse can be utilized to support fast finance dashboard queries.

MAXIMIZE THE UTILIZATION OF SNOWFLAKE CACHE

While segmenting query workload, users querying the same data can be placed in the same virtual warehouse. This maximizes the chance of data retrieved to the cache by one user to be used by others.

Users shouldn't be too hasty while suspending a non-functional virtual warehouse. The default setting automatically suspends any warehouse after 10 minutes and auto-resumes functions with the execution of a SQL statement. The auto-suspend function can be set to as little as a few seconds for saving money but when resumed, the virtual warehouse cache will be clean, losing the performance benefits of caching.

The result cache is independent of the virtual warehouse, and any query executed by a user on the account will be from the result cache, with the SQL text being the same.

SCALE UP TO ACCOMMODATE LARGE WORKLOADS

The warehouse size can be swiftly and conveniently adjusted on Snowflake, with its pool of available resources. With the processing being over, the cluster can be automatically suspended, immediately. Upon requirement, the cluster resumes automatically when another query is executed. It's an entirely transparent process till the end-user application.

Local storage is always fast SSD in virtual warehouses and any large sort operations that can't be completed in memory will spill over to local storage. When a large amount of data is spilled over to external storage, it indicates that even the SSD storage is full, and the data is written to S3 or Blob storage, making it slower. These are clear indicators of shifting to a larger virtual warehouse with more memory and local SSD storage.

SCALING OUT FOR CONCURRENCY

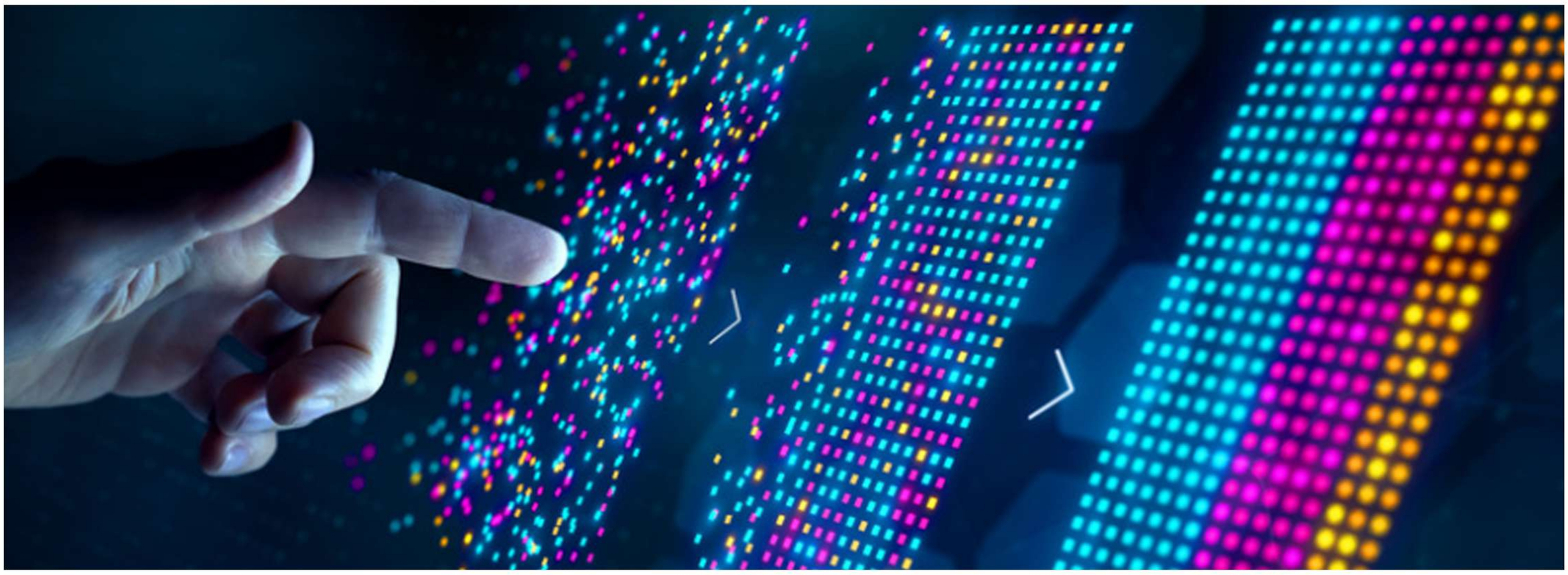
In contrast to the scaling-up option described above, this technique deploys additional clusters of same-sized nodes to achieve concurrency, a feature that increases the number of users and not the task size or complexity levels.

With varying numbers of concurrent users throughout the day, the clusters get suspended automatically and the users are only charged for the compute resources needed.

DATA CLUSTERING

The huge tables, over 1 terabyte in size, require designers to define a cluster key to maximize query performance. A cluster key maximizes partition elimination, improving query performance.

With data being loaded by date, it gets naturally clustered, where all data for the same day falls into the same micro-partition. Upon a need arising, a background task will automatically re-cluster the data and this compute processing will become a separate item. As Snowflake has the minimum and maximum value of every column in every micro-partition, it skips the micro-partitions that don't meet the query criteria.



WAREHOUSE OPTIMIZATION FOR PERFORMANCE

Performance Tuning Requirement	Strategy
Lower Queues	Performance can be improved by minimizing the queues, reducing the time taken between a query submission and getting the results.
Memory Spillage Issue Resolution	Upon the warehouse running out of memory, queries run significantly slower, resulting in byte spillage onto storage. Available memory adjustment improves performance.
Larger Warehouse	More compute resources would be available for query execution on a larger warehouse.
Query Acceleration	This process offloads parts of the query processing to serverless compute resources, speeding up query processing while reducing its demand on the compute resources.
Warehouse Cache Optimization	There is an improvement in query performance if it can be read from the warehouse's cache instead of the tables.
Limited Running of Concurrent Queries	Restricting the number of concurrent queries improves the performance as there are fewer queries demanding warehouse resources.

Conclusion

Snowflake is a secure, scalable, and popular cloud data warehousing solution. With the tips above, there is a better chance of harnessing the most from Snowflake. Scaling up to a bigger data warehouse is not a solution to all problems. The solutions are multifaceted and depend on factors such as end-user queries and transformation. This does not imply that scaling up is not an effective strategy. It is suitable when there is a need to improve query performance. However, before selecting the option of scaling up, it's advised to locate the issue affecting the performance of the data warehouse. Although there are multiple performance tuning options on Snowflake, it is essential to know the best practices that can be used to boost query performance and throughput.

UNITED TECHNO

united we solve

www.unitedtechno.com

info@unitedtechno.com