2024년 10월 10일

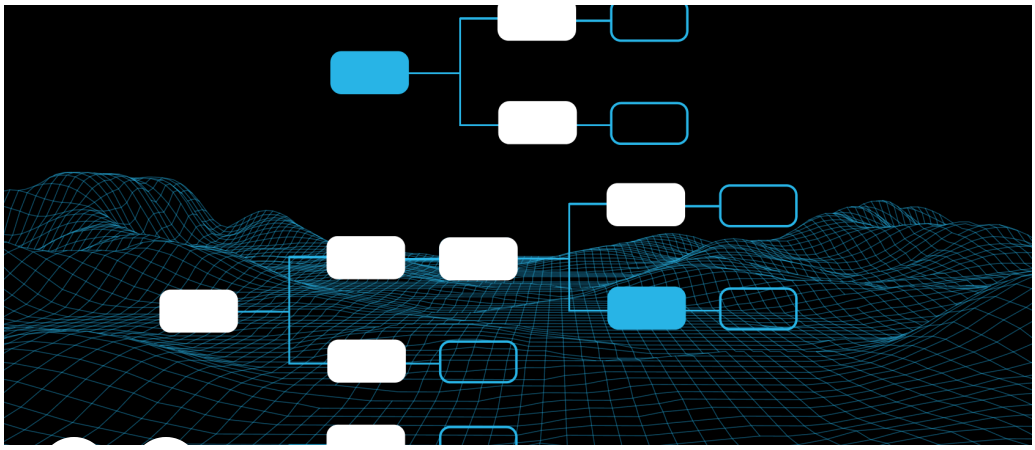
AUTHOR

Sahaj Saini

Rishabh Ahluwalia

Snowflake Speeds Up Cloning of Large Databases By 3x On Average — Here's How We Did It

Core Platform



SHARE

Have you ever had to wait hours to spin up a copy of your production database so you could have a test or a development environment? And had to pay extra for the test or development environment to be able to hold all the data?

Enter Snowflake Cloning — one of Snowflake's most popular features that clones databases, schemas and tables without physically copying any data. This means you can make your data available almost instantly for multiple teams, without compromising costs or adding loads of time to duplicate data. When Snowflake launched zero-copy cloning several years ago, it felt like pure magic.

Cloning very quickly became the new status quo. Today, fast cloning is more than just a convenience — it's essential. For example, a large European retailer relies on database clones for its CI/CD workflows — dozens of developers spin up hundreds of ephemeral sandbox environments, which are clones of the production database, every morning for their development and testing tasks. A healthcare provider delivers nightly snapshots of data to consumers through schema clones to speed up delivery.

The challenge and how Snowflake addresses it

Snowflake's architecture is divided into 3 layers: Cloud Services, Query Processing and Storage. When you create a table in Snowflake, its metadata (like table names and column definitions) is stored in the Cloud Services layer, while the actual data is stored as immutable files called micro-partitions in the Storage layer.

AUTHOR

Sahaj Saini

Rishabh Ahluwalia

When you clone a table, Snowflake doesn't physically copy the data. Instead, it quickly creates a new table by duplicating the metadata. Both the original and cloned tables reference the same underlying data files, but from that point on, each table can be modified independently without affecting the other.

This powerful cloning feature isn't limited to just tables; you can clone entire schemas or databases in the same way. By copying the metadata of all objects (tables, views, procedures, etc.), Snowflake allows you to create fast, independent snapshots.

```
CREATE DATABASE clone_important_db CLONE
important_db;
```


As customers scale their usage of Snowflake, the stakes rise dramatically. The number of objects in databases and schemas can skyrocket from just a few to tens of thousands. This explosion of metadata can turn what used to be fast clone operations into time-consuming bottlenecks. For example, the aforementioned healthcare provider saw its schema cloning times balloon to more than 35 minutes — a serious drag on its nightly processes to deliver snapshots to consumers.

To address this, we introduced a new cloning optimization that spins up more resources and parallelizes the metadata-copying process. With this optimization, we've slashed cloning times for even the most metadata-heavy databases and schemas. While we would allocate more resources, the overall time is cut so that the net cost remains comparable.

This optimization means faster clone operations, no matter how big your database or schema grows, and it keeps your teams moving at full speed.

Results

Cloning optimization is now enabled by default on all accounts in all Snowflake regions. We built some dashboards to continuously monitor its impact.

We compared stable database clone operations before and after the rollout to study the impact of the optimization (Figure 1).

AUTHOR

Sahaj Saini

Rishabh Ahluwalia

Figure 1. Database clone execution time, before and after optimization

The Source DB Size Category is defined in terms of Table count, as follows:

Category 0: Table count < 100

Category 1: Table count between 100 and 1000

Category 2: Table count greater than 1000

The median clone execution time improved by 12% for “small” databases; 41% for “medium” databases; and 82% for “large” databases. The average improvement for large databases was 3x.

The healthcare provider mentioned earlier saw its schema-cloning time plummet from more than 35 minutes down to just five — a 7x improvement. This enhancement was delivered seamlessly to customers across our entire fleet, providing transparent performance gains without extra effort. The result is continuously improving economics and performance for all Snowflake workloads.

Conclusion

At Snowflake, we’re on a continuous quest to enhance performance, with a particular focus on accelerating the core database engine, and we are proud to deliver these performance improvements through our weekly releases. In this blog post, we covered a recently released performance optimization that’s broadly applicable, highly impactful and now generally available to all customers.

To learn how Snowflake measures and prioritizes performance improvements, please read more about the Snowflake Performance Index here. For a list of key performance improvements by year and month, visit Snowflake Documentation.

RELATED CONTENT

2024년 07월 30일

Adaptive Network Optimizations for Faster Query Performance

At Snowflake, we strive to deliver “automatic performance” to all our customers. This performance is driven through multiple areas of investment: hardware-level optimization, intelligent resource allocation, proactive storage optimization, adaptive…

Find Out More

2024년 09월 06일

Aggregation Placement — An Adaptive Query Optimization for Snowflake

Snowflake’s Data Cloud is backed by a data platform designed from the ground up to…

Discover

2024년 09월 06일

Benchmarking Real World Customer-Experienced Performance Using the Snowflake Performance Index (SPI)

I’m excited to share some details about one of the projects that I’ve been working…

Delve into the details

Snowflake Inc.

AUTHOR

Sahaj Saini

Rishabh Ahluwalia