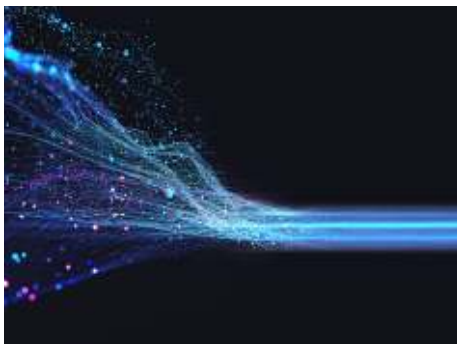
MICRO FOCUS | VERTICA

Customer Case Study

Analytics and Big Data



Why a major global brand chose Vertica over Snowflake

A mini-case study based on a side-by-side product comparison: Vertica vs. Snowflake

Overview

With a significant presence in professional markets worldwide, a well-known website gathers a massive amount of data related to its many products and services. In order to provide meaningful analytics to its global customer base, the company relies on data storage and management technology – an analytics database – that supports many thousands of concurrent customer interactions. Because the performance of the technology directly affects the customer experience on the website, it is vital that the database provide rapid responses without errors.

The company's new analytics experience is built with consistency and usability in mind. The response time for queries needs to be in the sub-second range in most cases, but the scale of the data driving the user experience is enormous and requires special big data technology in order to satisfy requirements and expectations.

Proof of Concept (POC) Details

With a need to improve the existing data analytics capability, which was based on Google Big Query, the company's engineering team decided to test two software-as-a-service (SaaS) solutions – Vertica Accelerator and another leading solution – to determine which would best satisfy its criteria around performance and concurrency.

The team devised several tests to simulate effective performance for a multi-product analytics and insights tool – considering both current and future needs. The team was primarily interested in response time, but another decision factor was the cost of the new system. The team identified seventeen queries to be used in the benchmarking exercise – ranging from requests submitted through the user interface to subqueries.

The engineering team tested both SaaS solutions using data volumes typical for daily traffic, as follows:

Customers (determined by the number of transactions under that customer)	Traffic Percentage
Small	98%
Larger	2%

Date Range	Traffic Percentage
Under 30 days of data	95%
Between 30-60 days of data	2%
Between 60-90 days of data	1%
More than 90 days of data	2%

Testing methods

The engineering team used Jmeter to run the queries concurrently, and conducted multiple tests to determine the performance of Vertica and Snowflake. Each test ran for one hour and consisted of 34 different queries running simultaneously, with small and large customers requesting data for different data ranges. The traffic request was modeled according to the breakdown shown in the tables above.

The tests were run using 200, 500, and 1,000 user connections to the database, with a ramp-up time of 200s, 500s, and 1,000s respectively, and a random connection delay of 0-50ms between queries to allow the database some natural breathing room to simulate normal operations.

A list of 2,700 customers were selected for these runs, of which 2,250 were classified as small customers and 450 as large customers. The testing cycled through these customers.

There was no set caching on the database, except for a filesystem caching mechanism that was already in place in the operating system.

Note: Due to the nature of Vertica's resource pools feature, the team ran the experiment multiple times.

Vertica Tests and Results

Vertica was tuned according to recommendations from the Vertica sales engineers. This involved running queries and sharing findings, then conducting further tuning as needed. Data was preloaded into the Vertica database prior to running the tests to leverage two key Vertica features: a Flattened Table to reduce the query overhead of dimension table joins, and multiple Live Aggregate Projections to optimize fact table aggregate metrics. The original recommendation was to run this on 12 nodes, but the company opted for 6 initially.

Testing involved three distinct "experiments," each comprising multiple tests for large and small customers, as follows:

- Experiment 1 - One general resource pool
- Experiment 2 - Separate resource pools
- Experiment 3 - Separate resource pools for small and large customers with a fixed 30TPS (transactions per second)

As expected, the time to complete requests increased as the number of users hitting the database increased, and as the data range increased. All requests completed with no errors.

Due to the nature of the traffic flow, data for queries with a data range of 90+ days were few in number and produced the largest latency due to the size of the data being returned.

The team saw that using separate pools for large and small customers improved latency, with the greatest improvement around the 1-second mark when all samples were

averaged. The team saw the greatest improvement in the 95th and 99th percentiles, where latency was almost halved.

Data was loaded such that all nodes were utilized and no node was overloaded. Monitoring of the database showed that the CPU on all nodes did not increase beyond 80% utilization for the most active nodes, with the least active node recording 60% utilization.

Engineering team recommendations

The company's engineering and product management teams were pleased with the results obtained from the Vertica solution, as the performance and concurrency metrics allow room for growth and the test included a truly representative mix of all queries being targeted. They recommended that a few more sets of tests be conducted, just targeting 95% of the traffic and data range to give a good indication of how the database would perform.

Tests and Results Snowflake

For the tests on Snowflake, the team ran the same workloads on a data warehouse schema loaded with the same exact data as used with Vertica. The tables and models were identical.

Because Snowflake does not offer live aggregate projections, the team used materialized views (physically created views) as a pre-aggregation medium instead.

The team ran two tests, both with workload management:

- 24 XS (extra small) warehouses as recommended by Snowflake Engineering
- 24 M (medium) warehouses utilizing an expanded materialized view by including more 'group by' columns.

With Snowflake, the team noticed some query failures. In addition, the response times for the extra small warehouse tests in the 90th and 95th percentile data ranges did not meet the criteria provided to the vendor's engineers. The following table compares query response times, in seconds, between Snowflake and Vertica.

	Data range	Snowflake	Vertica
200 users	90th Percentile	25.77 sec.	2.10 sec
	95th Percentile	34.76 sec	6.11 sec
500 users	90th Percentile	34.98 sec	2.69 sec
	95th Percentile	51.91 sec	7.35 sec

Pricing comparison: Vertica vs. Snowflake

Snowflake

Because the testing could not determine how many resources would be required to meet the company's criteria, pricing has not yet been finalized with the vendor. However, the engineering team reported, "The 24 warehouses we used in the experiment will cost around $515,088 (budgetary number with discounts and including the compute price). This solution still doesn't meet the criteria and we will require more resources, which will make the final price higher."

Vertica

The cost for Vertica Accelerator is straightforward: pricing is based on an hourly fee per CPU. The more hours/CPUs purchased in advance, the lower the price. The base price for one hour on one CPU is $0.09 not including the compute cost (the solution is based on AWS EC2 resources). The Accelerator program allows the infrastructure to be owned by the client and hosted inside a private cloud maintained by Vertica. Depending on the chosen cluster size, the cost will be between $140,777 and $193,000.

The client will incur the cost of the AWS resources used to spin the Vertica cluster; the estimated compute cost is $72,849.72. Furthermore, Vertica will provide one week of free professional services with the purchase of two weeks of that same service.

Other Considerations

The company's engineering team reported several other details that affected the quality of their engagement with Vertica and Snowflake during the course of evaluation:

- With both vendors, support for infrastructure and software were included.
- Working with both vendors on the POCs, the response and engagement from Vertica were much better than Snowflake.
- Vertica required very little time to dedicate sales and engineering resources. Snowflake took 5 weeks to dedicate a resource to work with the team on the POC.
- Both vendors showed a high level of competence and knowledge of their own software and how it should fit the use case at hand.
- With Vertica, the POC was time-boxed and successfully completed in 2 weeks, which compares favorably to 6 weeks taken so far by Snowflake without achieving the criteria.
- Snowflake provides resident engineering support at $134K a year.
- Both vendors provide professional service on a prepaid-hours model:
  - Vertica: $12,500 for a week (40 hours) of a remote consultant.
  - Other vendor: $15,500 for a bucket of 40 hours

Conclusion

Snowflake was already in use in other areas of the business, so some teams already had in-house knowledge of the technology; by contrast, Vertica was brand new to them. However, even given the company's established knowledge of the Snowflake product, it still couldn't beat Vertica. This head-to-head competition is a great example of the blazingly fast performance enabled by the query optimization strategies native to Vertica's architecture and approach to analytics. With a few easily made customizations (using Flattened Tables and Live Aggregate Projections), Vertica was able to satisfy the query response time requirements using fewer compute resources and at lower financial cost than Snowflake. Furthermore, as the numbers in the above table indicate, Vertica query execution was a massive 6 to 12 times faster.