



William McKnight, Jake Dolezal
Jun 14, 2022

Transactional and Analytical Workloads

How Transactional and Analytical Performance Impacts the TCO of Cloud Databases

Data Infrastructure, AI & Analytics

Transactional and Analytical Workloads

How Transactional and Analytical Performance Impacts the TCO of Cloud Databases

Table of Contents

- 1 Summary
- 2 Data Architecture Selection and TCO
- 3 Platform Summary
- 4 Performance Testing
- 5 TCO Testing and Analysis
- 6 Conclusion
- 7 Disclaimer
- 8 About SingleStore
- 9 About William McKnight
- 10 About Jake Dolezal
- 11 About GigaOm
- 12 Copyright

1. Summary

Competitive data-driven organizations rely on data-intensive applications to win in the digital service economy. These applications require a robust data tier that can handle the diverse workloads demands of both transactional and analytical processing while serving an interactive, immersive customer experience. The resulting database workloads demand low-latency responses, fast streaming data ingestion, complex analytic queries, high concurrency, and large data volumes.

Typically, organizations solve this problem by using several database technologies stitched together, but this increases development complexity, operational costs, and data duplication. SingleStoreDB is a cloud-native distributed relational database that unifies transactional and analytical processing in a single database technology, which simplifies the application data infrastructure and allows businesses to innovate and deliver their digital services more quickly.

Businesses need modern scalable architectures and high levels of performance and reliability to gain timely analytical insights and earn competitive advantage. They also value fully managed cloud services that can leverage powerful data platforms without the technical debt and burden of finding talent to manage the resources and architecture in-house. These as-a-service models enable users to only pay as they play and to stand up a fully functional analytical platform in the cloud with just a few clicks.

This report outlines the results from a GigaOm Field Test derived from three industry standard benchmarks—TPC Benchmark™ H (TPC-H), TPC Benchmark™ DS (TPC-DS), and TPC Benchmark™ C (TPC-C)—to compare **SingleStoreDB**, **Amazon Redshift**, and **Snowflake**. We found that SingleStoreDB offers benefits that include:

- **50% Savings** over three years compared to Snowflake-MySQL stack
- **60% Savings** over the same period compared to Redshift-PostgreSQL stack
- **Up to 100% Faster** in TPC-H workloads compared to Redshift (with Refresh)

Note that these standard benchmarks for transactions and analytics mostly address data at rest. By contrast, the behavior of a database system under load—performing streaming data ingestion while processing transactions, computing analytics, and serving users concurrently—is often defined more by data in motion.

For operational workloads we used SingleStoreDB as a rowstore, while also using it as a columnstore for comparison across both transactional and analytic tests. Two issues are relevant here: First, columnstore is generally not recommended for transactional workloads but is applied here for comparative insight; and second, the columnstore implementation in SingleStoreDB is called Universal Storage and adds capabilities like fast point reads, updates, and deletes. It is significantly more performant and effective in analytic scenarios than traditional columnstore approaches.

These tests produced interesting results that reveal some of the performance characteristics of these platforms. The tests compared the performance of the following fully managed cloud database-as-a-service offerings:

- A MySQL, Snowflake, ETL, Redis, and Kafka Stack
- A PostgreSQL, Redshift, ETL, Redis, and Kafka Stack
- A SingleStoreDB and Kafka Stack

The results of the GigaOM Analytical Field Test are valuable to all analytical functions within an organization, addressing scenarios such as data analysis, operational reporting, interactive business intelligence (BI), data science, machine learning, and more.

The results of the GigaOM Transactional Field Test, meanwhile, are valuable to all operational functions of an organization, such as human resource management, production planning, material management, financial supply chain management, sales and distribution, financial accounting and controlling, plant maintenance, and quality management.

We also tested SingleStoreDB, Redshift, and Snowflake with a 10TB TPC-H-like analytical workload. From the configurations we tested, we found that an S-48 SingleStoreDB cluster was slightly faster than a Snowflake 3X-Large cluster and more than twice as fast as a 10-node ra3.16xlarge Redshift cluster.

We then calculate the annual costs of the platform stacks and the time-effort costs (people costs, development costs and production costs) to conclude that SingleStoreDB is 2x cheaper than the Snowflake-MySQL stack and 2.5x cheaper than the Redshift-PostgreSQL stack over three years running enterprise-equivalent workloads. What's more, as shown in **Figure 1**, only SingleStoreDB is able to handle both analytical and transactional workloads in the same engine.

Our final conclusion: The **superiority in transactional processing** and the **high competitiveness in**

analytic processing makes SingleStoreDB worth consideration as a single database solution that addresses a spectrum of enterprise workloads.

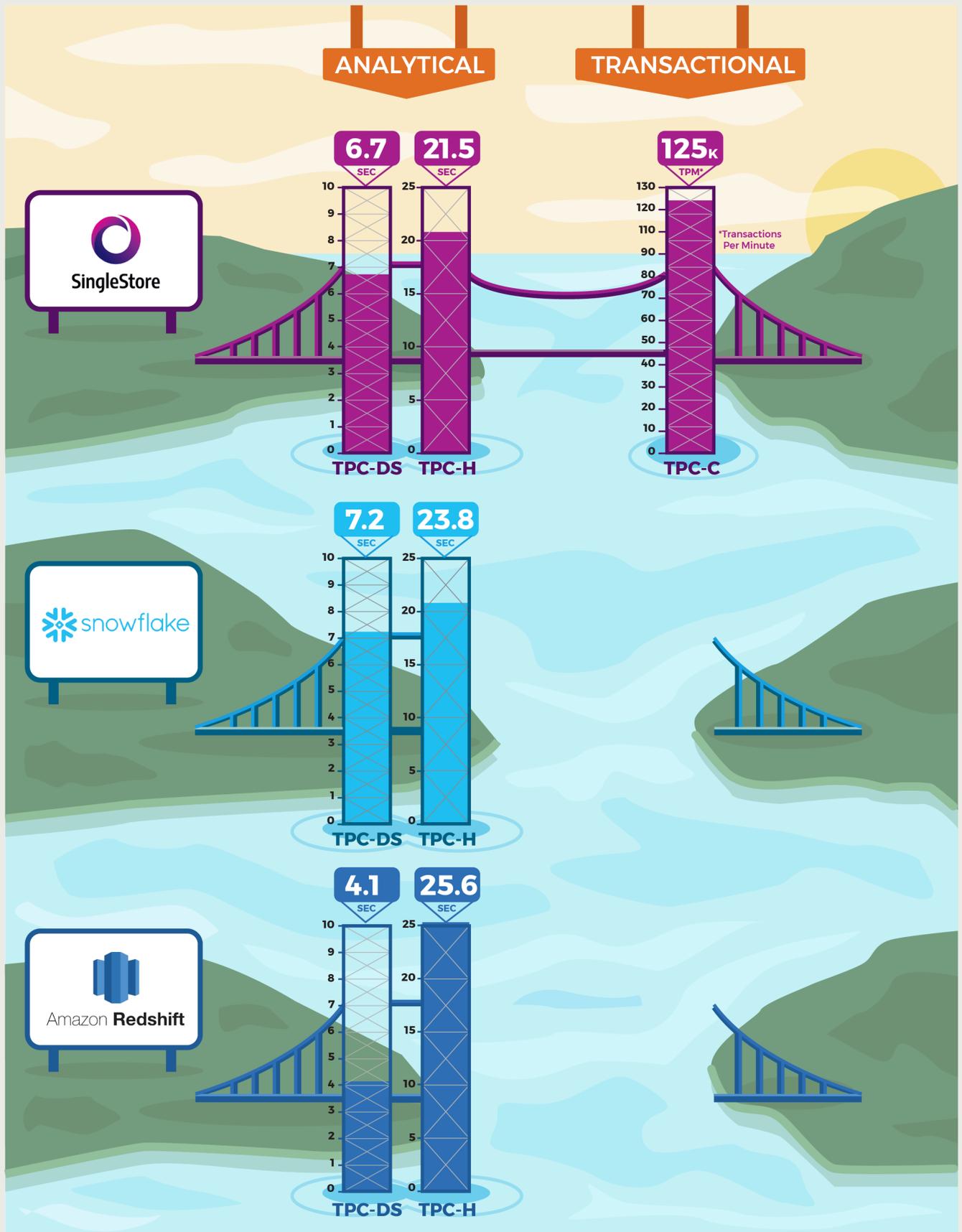


Figure 1. Bridging Workloads: Database Performance Across Analytical and Transactional Workloads

As shown in **Figure 1**, the results of the TPC-DS-like and TPC-H-like tests are measured in seconds, with SingleStoreDB leading both Redshift and Snowflake in TPC-H-like performance, and edging out Snowflake in TPC-DS-like performance. The right side shows the ability of SingleStoreDB to run transactional workloads (such as those in the TPC-C-like benchmark) when other solutions cannot.

2. Data Architecture Selection and TCO

Organizations run their day-in-and-day-out businesses with transactional applications and databases. On the other hand, organizations glean insights and make critical decisions using analytical databases and business intelligence tools.

The transactional workloads are relegated to database engines designed and tuned for transactional high throughput. Given that transactions happen at all hours and from a variety of sources, the speed of transactional data creation has soared. Whether it is processing financial transactions, processing video game activities, or monitoring health conditions, the data is created in high volumes at high speed. Latency—the time it takes to access or write—delays action. Thus, latency is particularly relevant to read- and write-intensive applications in the cloud, where database latency, plus latency created by network throughput, API/microservices calls, and other processes compound.

Meanwhile, the big data generated by all the transactions require analytics platforms to load, store, and analyze volumes of data at high speed, providing timely insights to businesses. Data-driven organizations leverage this data, for example, for advanced analysis to market new promotions, operational analytics to drive efficiency, or for predictive analytics to evaluate credit risk and detect fraud. Customers are leveraging a mix of relational analytical databases and data warehouses to gain analytic insights.

Thus, in conventional information architectures, this requires two different database architectures and platforms: online transactional processing (OLTP) platforms to handle transactional workloads and online analytical processing (OLAP) engines to perform analytics and reporting. These architectures mandate investment and effort in performing the data transformations and data movement required to move data from one engine to the other, usually using ETL/ELT processing and batch windows.

There are a variety of databases available to the transactional application. Ideally, any database would have the required capabilities; however, depending on the scale of the application, and the architecture of the database solutions, not all databases can perform without massive hardware scale-up and its associated capital outlay. Likewise, there are many database platforms that are designed and touted for purposes, such as data warehousing, analytics, machine learning, and so forth. These databases are not designed for high volume, high throughput transactional workloads.

These trends force organizations to make often difficult architecture decisions to manage the cloud and all their workloads. The choice of a platform under one cloud provider may have ripple effects in subsequent platform decisions. For instance, if one chooses a transactional database within one

cloud ecosystem will likely lean to the analytical offering with that vendor as well to avoid over complexity and expensive data movement (both in terms of network latency and cloud egress charges). Organizations may find themselves hemmed in once an initial selection is made—even if subsequent platform choices are less attractive than alternatives from other vendors.

Finally, what about the increasing need and use cases for operational analytics? Today, modern SaaS applications demand interactive analytics on fast-moving operational data. Also, operational analytics is NOT a data warehouse of stale data loaded in batch the previous day; rather it is real-time analysis as-is with no data movement necessary. In this scenario, instead of analyzing data within a single system, organizations desire to pipe the data from many disparate systems, machines, applications, and devices into one place and analyze it on the fly for up-to-the minute insights or train and deploy machine learning algorithms for automated decision making. Organizations are often reluctant to attempt analyzing real-time data, fearing the analytical workload will hamper the performance of the operational work that has to be the priority. Rightfully so, since as previously stated, most databases are designed for one type of workload or the other—transactional or analytical—but not both.

3. Platform Summary

Both operational and data analytics platforms load, store, and analyze volumes of data at high speed, providing timely insights to businesses. Data-driven organizations leverage this data for advanced analysis to market new promotions, operational analytics to drive efficiency, or for predictive analytics to evaluate credit risk and detect fraud, for example. Customers are leveraging a mix of relational analytical queries on transactional databases to gain analytic insights.

This report focuses on both transactional and analytical databases in the cloud, because deployments are at an all-time high and poised to expand dramatically. The cloud enables enterprises to differentiate and innovate with these database systems at a much more rapid pace than was ever possible before. The cloud is a disruptive technology, offering more elastic scalability, enabling faster server deployment and application development, and allowing less costly storage vis-à-vis on-premises deployments. For these reasons and others, many companies have leveraged the cloud to maintain or gain momentum as a company.

This report compares SingleStoreDB as an all-in-one database to Amazon Redshift and Snowflake with two analytical workloads and with a transactional workload. Our purpose was to ascertain whether SingleStoreDB—as a multiple workload platform—performs at a level comparable to popular cloud databases.

SingleStoreDB

SingleStoreDB is a next-gen, cloud-native distributed relational database designed to power today's data-intensive applications. It is designed to match the performance of specialized databases for both transactional (OLTP) and analytical (OLAP) workloads in a single unified engine to drive maximum performance for your modern applications. The selected tests TPC-C, TPC-H, and TPC-DS were each chosen with a view to illustrate SingleStoreDB's ability to perform across the range of workloads for operations and analytics. With SingleStoreDB, customers can ingest millions of events per second with ACID transactions using SingleStoreDB Pipelines, while simultaneously delivering low-latency SQL queries on billions of rows of data.

With patented **Universal Storage**, SingleStoreDB allows both transactional and analytical workloads to be processed using a single table type. Universal Storage is based on the traditional columnstore, with the added benefit of fast point reads, updates, and deletes—operations that are usually poorly supported in columnstores. Throughout the document, where columnstore is referenced, it is Universal Storage.

This together with separation of storage and compute, Point-In-Time Recovery (PITR) and new workspaces, SingleStoreDB can be the single database for all data-intensive applications and workloads. For testing, we utilized the current version of the SingleStoreDB Cloud, which is the fully-managed, cloud database-as-a-service running on AWS.

Amazon Redshift

Amazon Web Services Redshift was the first managed data warehouse cloud service and continues to get a high level of mindshare in this category. It indeed ticks all the table stakes boxes for a cloud analytic database. Amazon Redshift is a fit for organizations needing a data warehouse with little to no administrative overhead and a clear, consistent pricing model.

We tested the latest RA3 Redshift engine that introduced a new managed storage layer, which is an upgrade from the tighter coupled storage on the older DS2 and DC2 instance types. For Redshift, we paid an hourly rate for when the cluster was running, but it also has a pause feature to stop billing.

Snowflake

As a cloud-only, fully managed solution, Snowflake has a clear separation between compute and storage. For Snowflake on AWS, which is what we used for the queries, data is stored in AWS S3 and is cached when queries are executed to bring the data in closer proximity to compute resources. Snowflake essentially offers two configuration “levers”—the size of the warehouse cluster and how many clusters are permitted to spin up to handle concurrency. Snowflake scales by cluster server count in powers of 2 (i.e., 1, 2, 4, 8, 16, and so on.) If enabled, Snowflake will spin up additional clusters to handle multi-user concurrent query workloads. Snowflake would automatically spin the additional clusters down once demand has passed. If not enabled, it will place paused queries in a queue until resources free up.

For Snowflake, there is a flat hourly fee for when hourly compute resources are being used. We tested \$2 per credit per hour for the Standard tier. Once the compute warehouse goes inactive, you no longer pay. However, there is a separate charge for data storage.

Table 1. Platform Summary

	SingleStoreDB	Amazon Redshift	Snowflake
Version Tested	7.8	1.0.34928	6.3.3
Testing Category	Transactional + Analytical	Analytical	Analytical

Source: GigaOm 2022

4. Performance Testing

Test Setup

The setup for this Field Test was informed by the [TPC Benchmark™ H](#) (TPC-H), [TPC Benchmark™ DS](#) (TPC-DS), and [TPC Benchmark™ C](#) (TPC-C) specifications. These are **NOT** official TPC benchmarks. The queries were executed using the following setup, environment, standards, and configurations.

Benchmark Data

The data sets used in the benchmark were a workload derived from the well-recognized industry standard benchmarks TPC-H, TPC-DS, and TPC-C. From tpc.org:

“The TPC-H is a decision support benchmark. It consists of a suite of business-oriented ad-hoc queries and concurrent data modifications. The queries and the data populating the database have been chosen to have broad industry-wide relevance. This benchmark illustrates decision support systems that examine large volumes of data, execute queries with a high degree of complexity, and give answers to critical business questions.

“TPC-DS is a decision support benchmark that models several generally applicable aspects of a decision support system, including queries and data maintenance. The benchmark provides a representative evaluation of performance as a general purpose decision support system. The purpose of TPC benchmarks is to provide relevant, objective performance data to industry users.

“TPC-C is an on-line transaction processing (OLTP) benchmark. TPC-C involves a mix of five concurrent transactions of different types and complexity either executed on-line or queued for deferred execution. The database is comprised of nine types of tables with a wide range of record and population sizes. TPC-C is measured in transactions per minute (tpmC). While the benchmark portrays the activity of a wholesale supplier, TPC-C is not limited to the activity of any particular business segment, but rather represents any industry that must manage, sell, or distribute a product or service.”

For our TPC-H-like and TPC-DS-like analytical workloads, we tested a single user Power Run with refresh/maintenance scripts running in the background. See the TPC specifications for more information, but in short, this is a series of data manipulation language (DML) statements—INSERT, UPDATE, and DELETE—that execute in a steady stream that constantly change the underlying data for the SELECT analytical queries.

To perform a TPC-C-like workload, most people use a driver that is already built. There are several TPC-C-like drivers available for free. To test SingleStoreDB, we used the [Percona Lab tpcc-mysql implementation](#)—open source and available on GitHub—which is a TPC-C-like test specifically for MySQL.

SingleStoreDB loads data very rapidly using its Pipelines feature. SingleStoreDB has a [fork of the Percona Labs tpcc-mysql](#), specifically to generate raw data on disk that we uploaded to AWS S3 and loaded into SingleStoreDB with Pipelines. We used this fork for generating the dataset on disk.

The SingleStoreDB fork also has a feature that allows you to run the TPC-C-like test with the maximum number of possible connections (in our case, 10,000), but divide them among drivers in equal percentages. We used this feature to test SingleStoreDB columnstore’s ability to a high concurrent workload and still perform well. We used the Percona Labs code base for the SingleStoreDB rowstore tests.

In summary, **Table 2** shows the testing we performed.

Table 2. Testing Summary

Category	Test	Scale	Execution	Platform
Transactional	TPC-C	10,000 warehouses	Percona Labs tpcc-mysql	SingleStoreDB (Rowstore)
Transactional	TPC-C	10,000 warehouses	SingleStoreDB fork tpcc-mysql	SingleStoreDB (Columnstore)
Analytical	TPC-H	10TB	Power run w/refresh streams	SingleStoreDB Redshift Snowflake
Analytical	TPC-DS	10TB	Power run w/maintenance	SingleStoreDB Redshift Snowflake

Source: GigaOm 2022

Database Environments

Our benchmark included the database environments shown in **Table 3** and **Table 4**:

Table 3. Analytical Configuration Summary

Analytical	SingleStoreDB	Snowflake	Redshift
Class/Size	S-48 384 vCPU 3TB RAM	3X-Large (64 nodes)	ra3.16xlarge (10 nodes) 480 vCPU 3.75TB RAM
Cost Per Hour	\$124.80	\$128.00	\$130.40

Source: GigaOm 2022

Table 4. Transactional Configuration Summary

Transactional	SingleStoreDB Rowstore	SingleStoreDB Columnstore
Class/Size	S-64 512 vCPU 4TB RAM	S-36 288 vCPU 2.25TB RAM
Cost Per Hour	\$166.40	\$83.20

Source: GigaOm 2022

*Single Availability Zone

For tuning the databases for optimal performance, we used the following methods.

For Redshift, we used Amazon Redshift [best practice recommendations](#) for designing tables. This included setting distribution keys and sort keys on key columns in the TPC-H and TPC-DS tables to optimize JOINS and data distribution. We also enabled the AQUA service. For Snowflake, we used cluster keys to also optimize JOINS and data distribution. We also enabled Search Optimization. For SingleStoreDB, we used clustered columnstore tables for the TPC-H-like and TPC-DS-like workloads.

We tested the TPC-C-like workload on SingleStoreDB with both rowstore and columnstore (Universal Storage) tables. Neither Snowflake nor Redshift support running TPC-C. As an example, unique constraints is a feature that is not supported in the data warehouses but is supported in SingleStoreDB. The TPC-H-like and TPC-DS-like workloads were executed against columnstore tables—the obvious choice for most analytical workloads. We wanted to see what the difference between rowstore and columnstore would be in a transactional workload to help make more informed data architectural choices.

Test Results

This section analyzes the results from the field tests we performed. **Figures 2** through 6 below show the results of these tests.

The TPC-H and TPC-DS graphs are a calculation of the [geometric mean](#) of the single-user power runs. The geometric mean finds the central tendency of the results and “smooths out” the effect of outlier execution times (either unusually high or unusually low). The formula used for the geometric mean is:

$$\left(\prod_{i=1}^n x_i \right)^{\frac{1}{n}} = \sqrt[n]{x_1 x_2 \cdots x_n}$$

Analytical Test (TPC-H-Like)

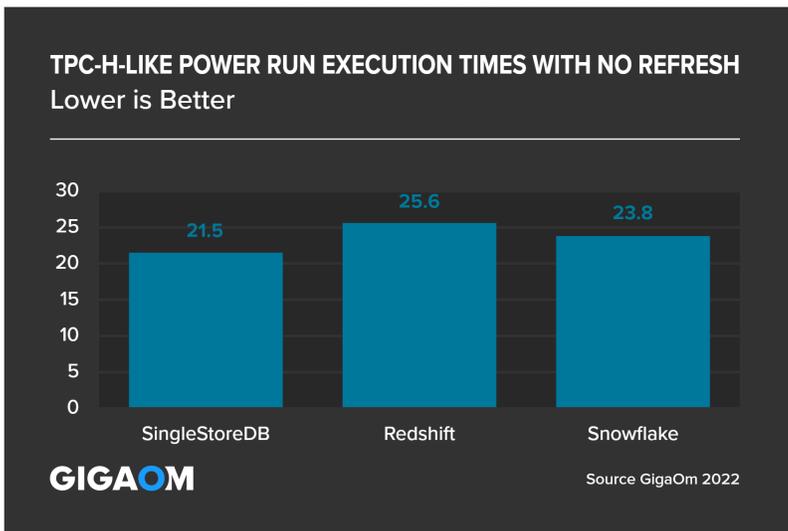


Figure 2. TPC-H-Like Power Run Execution Times with No Refresh (Lower is Better)

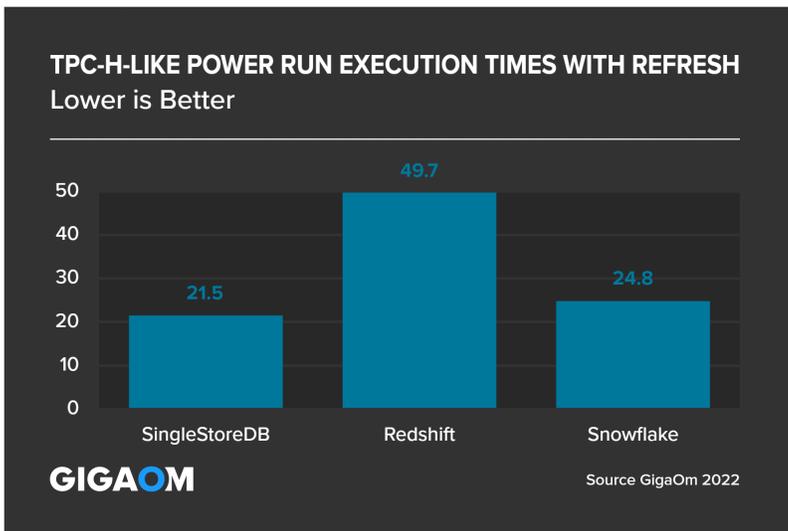


Figure 3. TPC-H-Like Power Run Execution Times with Refresh (Lower is Better)

Analytical Test (TPC-DS-Like)

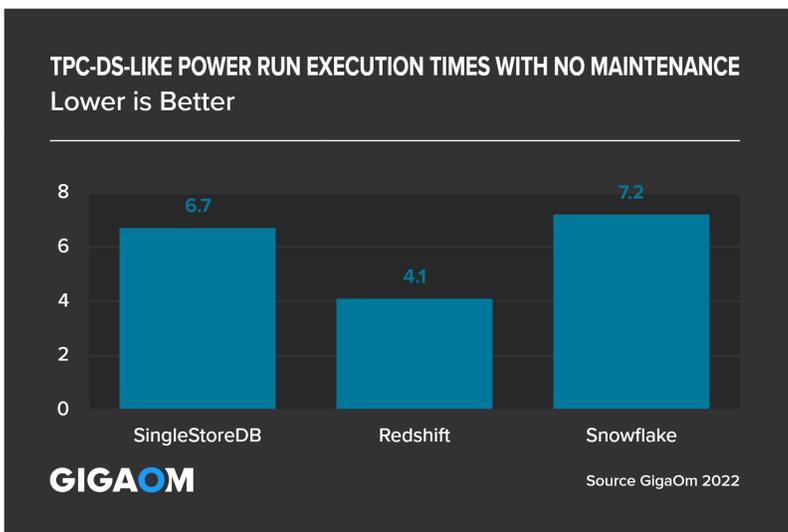


Figure 4. TPC-DS-Like Power Run Execution Times with No Maintenance (Lower is Better)

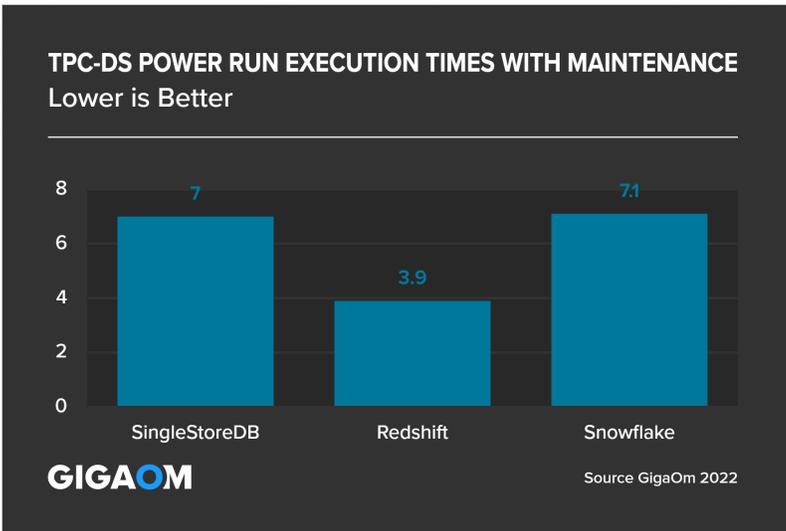


Figure 5. TPC-DS Power Run Execution Times with Maintenance (Lower is Better)

Transactional Test (TPC-C-Like)

As noted earlier, the TPC benchmarks we run are modified. The TPC-C benchmark, for example, throttles the workload so there is a maximum throughput for a given number of warehouses. To derive insight for this report, we ran the test with this artificial throttling disabled. The upshot is that the TPC-C-like tests published here are not comparable to other TPC-C results you may find on the web, as almost all of those will be throttled.

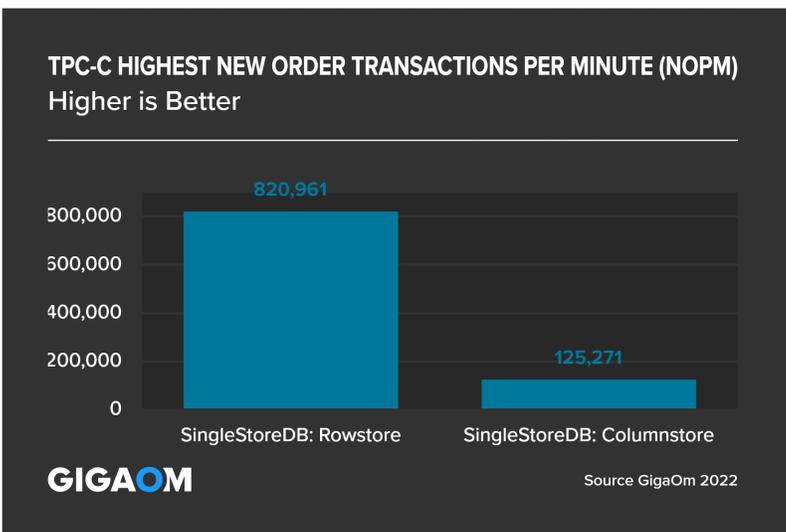


Figure 6. TPC-C Highest New Order Transactions per Minute (NOPM) (Higher is Better)

5. TCO Testing and Analysis

TCO Test Setup

Calculating the total cost of ownership in projects is something that happens formally or informally for many enterprise programs. It is also occurring with much more frequency than ever. Sometimes, well-meaning programs will use TCO calculations to justify a program but the measurement of the actual TCO on the flip side can be a daunting experience, especially if the justification TCO was entered into lightly.

This paper will focus on comparing the costs of a core stack of heterogeneous cloud data infrastructure to a single SingleStoreDB Cloud offering.

Modern applications today require a combination of operational, analytical, streaming, and caching capabilities to provide interactive data experiences. The categories, or components in these stacks, that we included in our TCO calculations are as follows:

- Transactional/Operational Database Compute and Storage
- Analytical Database Compute and Storage
- Cache Database Compute
- Data Integration/ETL
- Streaming Data Ingest and Delivery

In our use case, we built three different core information management stacks, depicted in **Figure 7**:

- A MySQL, Snowflake, ETL, Redis, and Kafka Stack
- A PostgreSQL, Redshift, ETL, Redis, and Kafka Stack
- A SingleStoreDB and Kafka Stack

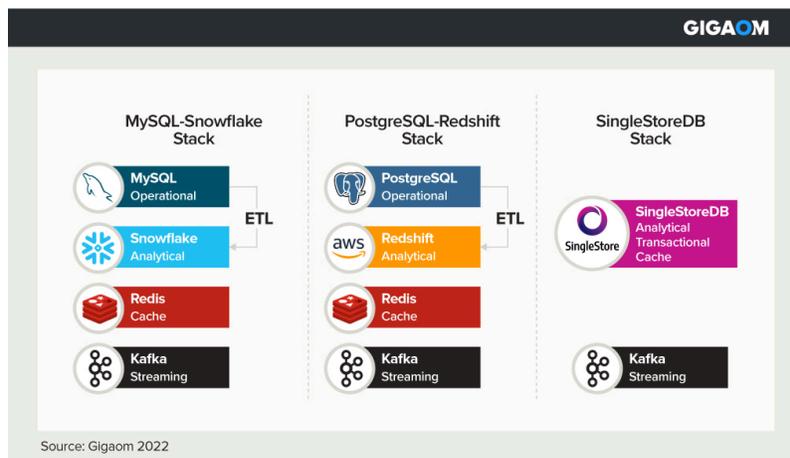


Figure 7. Core Information Management Stacks

In assembling these stacks, we made these assumptions:

- Redshift and PostgreSQL were paired together because they use the same SQL syntax, which we think is advantageous for development and administration purposes.
- The cache workload only represents 5% of our workload by CPU-memory load. Because Redis and SingleStoreDB both have in-memory capabilities, we assumed they have equivalent performance, especially since it is a small slice of the performance pie and represents a negligible +/- error either way.
- The same open-source Kafka was used across all three stacks.
- No data integration/ETL is needed with SingleStoreDB, since all the operational and analytical data are already co-located on the platform.

In addition to these components, other cost factors were considered. We realize that it takes more than simply buying cloud platforms and tools—namely, people—to achieve a production-ready enterprise stack. These resources are required to build up and maintain the platform to deliver business insights and meet the needs of a data-driven organization.

The additional people time-effort cost factors are considered in phases:

- Development Phase (build and/or migrate for Year 1)
- Production Phase (on-going maintenance and continuous improvement for Years 1, 2, and 3)

We combined all these costs to finally arrive at our final **three-year total cost of ownership** figures for the study.

Performance Equivalence

In order to create a use-case scenario to give us a basis for our pricing calculations, we assessed an approximated equivalent performance. To compare these stacks and size the infrastructure appropriately for equivalent performance, we used the results of our recent transactional and analytical field tests of SingleStoreDB, Redshift, Snowflake, PostgreSQL, and MySQL.

We tested SingleStoreDB, Redshift, and Snowflake with a 10TB [TPC-H-like](#) analytical workload. From the configurations we tested, we found that an S-48 SingleStoreDB cluster was more than twice as fast as a 10-node ra3.16xlarge Redshift cluster, and was 13% faster when compared to a Snowflake 3X-Large cluster.

We also tested SingleStoreDB (both with a rowstore and columnstore data orientation) with a 10,000 warehouse [TPC-C-like](#) transactional workload. SingleStoreDB rowstore provided more than a 6x throughput advantage over SingleStoreDB columnstore. Note that the SingleStoreDB columnstore implementation is an extension of traditional columnstores. It has deliberate modifications to enable point reads, writes, and updates at speeds significantly faster than traditional columnstores, without sacrificing the fast scan performance expected of a typical columnstore.

We used the performance results from that study to size our stacks for approximate equivalent performance, as shown in **Table 5**:

Table 5. Stack Sizing Table

-	SingleStoreDB	Snowflake	Redshift
Compute	S-64	3XLarge	ra3.16xlarge
Compute Units	64	64	20
Cost \$/unit/hour	\$2.60	\$2.00	\$13.04
Cost \$/hour	\$166.40	\$128.00	\$260.80
Cost \$/year	\$1,457,664.00	\$1,121,280.00	\$2,284,608.00

Source: GigaOm 2022

We arrived at these configurations by making the following assumptions:

- We tested SingleStoreDB S-32, S-48 and S-64 configurations. We selected the larger S-64 size for our stack, because it will be handling the operational, analytical, and cache workloads.
- SingleStoreDB S-48 and Snowflake 3X-Large have similar performance results, so we left Snowflake at the 3X-Large size.
- On Redshift, 10 nodes of ra3.16xlarge was twice as slow as Snowflake 3X-Large, so we assumed for the purposes of this study that 20 nodes would be adequate to achieve equivalent performance.

TCO Results

Running the test resulted in the following operations per second over a seven-day period:

Platform Costs

Table 6 shows our calculation of the annual costs for each of our platform stacks:

Table 6. Base Platform Costs per Year

	SingleStoreDB	Snowflake-MySQL	Redshift-PostgreSQL
1a) Transactional Compute		\$1,686,055	\$1,575,486
1b) Transactional Storage		\$18,072	\$18,072
2a) Analytical Compute		\$1,121,280	\$2,284,608
2b) Analytical Storage		\$4,800	\$2,949
3) Cache Compute		\$230,423	\$230,423
4) Integration/ETL Compute		\$30,835	\$30,835
5a) Transactional+Analytical+Cache Compute (No ETL Needed)	\$1,457,664		
5b) Transactional+Analytical+Storage	\$3,391		
6a) Streaming Analytics	\$13,140	\$13,140	\$13,140
6b) Streaming Ingest	\$145,766	\$145,766	\$145,766
6c) Streaming Delivery	\$134,554	\$134,554	\$134,554
GRAND TOTAL	\$1,754,515	\$3,384,925	\$4,435,833
Cost Difference	0%	93%	153%

Source: GigaOm 2022

To arrive at these figures, we made the following assumptions:

- Compute sizes were based on the approximate equivalent performance calculations discussed in the previous section.
- Analytical database storage costs for Redshift and Snowflake are based on 10TB of managed storage.
- Cache database costs assume using fully managed Redis on Amazon ElastiCache with four instances of cache.rg6.16xlarge with multi-region running 24/7/365.
- Data integration costs assume using the AWS Glue service with nightly ETL jobs running for three hours consuming 64 DPUs.
- SingleStoreDB managed storage costs are priced based on 12TB of storage (10TB for analytical and 2TB for operational).

- Kafka pricing is based on using the Confluent Cloud managed service, including the base service rate, 128GB per hour of write throughput, and 256GB per hour of read throughput.

Time-Effort Costs

In addition to these components, other cost factors were considered. We realize that it takes more than simply buying cloud platforms and tools—namely, people—to achieve a production-ready enterprise stack. These resources are required to build up and maintain the platform to deliver business insight and meet the needs of a data-driven organization.

The additional people time-effort cost factors are considered in phases:

- Development Phase (build and/or migration for Year 1)
- Production Phase (on-going maintenance and continuous improvement for Years 1, 2, and 3)

People Costs

To calculate TCO, one cannot overlook the people factor in the cost. It is people who build, migrate, and use the analytics platform. To figure labor costs for our TCO calculations, we use a blended rate of internal staff. **Table 7** provides our estimate of labor costs for the cost categories based on our industry experience:

Table 7. Labor Costs

-	Internal Staff
Average Annual Cash Compensation	\$125,000
Burden Rate	22%
Fully Burdened Annual Labor Cost	\$152,500
Standard Work Hours per Year	2,080
Hourly Rate	\$73
% Mix of Labor Effort in Production	100%
Source: GigaOm 2022	

For internal staff, we used an average annual cash compensation of \$125,000 and a 22% burden rate,

bringing the actual cost to a blended \$152,500. We also estimated the year to have 2,080 paid hours, which gives us an effective hourly rate of \$73 to support both the integration and migration to the platform and the on-going maintenance and continuous improvement.

Development Phase

To calculate the costs to build and/or migrate from an existing platform to the new architecture, we used the calculations in **Table 8**.

Table 8. Development Costs (Year 1 Only)

	SingleStoreDB	Snowflake-MySQL	Redshift-PostgreSQL
Source Objects	1,000	1,000	1,000
Modeling/Mapping Effort (hrs per object)	2	2	2
Complexity Multiplier	1	3	2.5
Total Hours	2,000	6,000	5,000
Reduction Due to POC Work	0%	0%	0%
Contingency Effort %	30%	30%	30%
Total Effort (Hours)	2,600	7,800	6,500
Average per Object	2.6	7.8	6.5
Labor Costs per Hour	\$73	\$73	\$73
Development Costs Total	\$190,625	\$571,875	\$476,563

Source: GigaOm 2022

Data migration includes significant data objects that need to be exported from the old platform and loaded into the new platform. Because SingleStoreDB is a fully-managed all-in-one platform, there is little-to-no development overhead for ETL and cross-platform integration. Thus, we used a complexity factor (or multiplier) of 1.0. For the other stacks, we considered all the integration, configuration, and tuning required, and gave them a complexity factor of 3.0 to demonstrate the time and effort required. However, we gave Redshift-PostgreSQL a 0.5 bonus for sharing nearly the same SQL syntax.

For the overall three-year TCO calculation, build phase costs will only be applied once during Year 1.

Production Phase

The cost of maintaining and improving the stacks has to be considered as well. This includes support work, such as database administration, disaster recovery, and security. However, no platform is (or should be) static. The environment needs constant improvement and enhancement to grow with the business needs. The work of project management and CI/CD integration are considered here as well.

The production labor costs detailed in **Table 9** were used in our calculations.

Table 9. Production Labor Costs

	SingleStoreDB	Snowflake-MySQL	Redshift-PostgreSQL
Transactional DBA FTE	0.25	0.25	0.25
Analytical DBA FTE	0	0.25	0.25
Streaming DBA FTE	0.25	0.25	0.25
Data Engineer	1	3	2.5
Total FTE	1.5	3.75	3.25
Production Costs Total	\$228,750	\$571,875	\$495,625
Source: GigaOm 2022			

With a fully-managed, homogenous platform like SingleStoreDB, administration and data engineering tasks are reduced.

Three-Year TCO

Finally, we arrive at our final three-year total cost of ownership figures for the study. **Table 10** provides a breakout by year for our stacks.

Table 10. TCO Breakdown by Year

	SingleStoreDB	Snowflake-MySQL	Redshift-PostgreSQL
Year 1 Platform Costs (Storage x1)	\$1,754,515	\$3,384,925	\$4,435,833
Year 1 People Costs	\$419,375	\$1,143,750	\$972,188
Year 1 Subtotal	\$2,173,890	\$4,528,675	\$5,408,021
Year 2 Platform Costs (Storage x2)	\$1,757,907	\$3,407,797	\$4,456,854
Year 2 People Costs (5% Increase)	\$240,188	\$600,469	\$520,406
Year 2 Subtotal	\$1,998,094	\$4,008,266	\$4,977,261
Year 3 Platform Costs (Storage x3)	\$1,761,298	\$3,430,669	\$4,477,876
Year 3 People Costs (5% Increase)	\$252,197	\$630,492	\$546,427
Year 3 Subtotal	\$2,013,495	\$4,061,161	\$5,024,302

Source: GigaOm 2022

Finally, **Figure 8** shows the grand total TCO comparison across the three stacks.

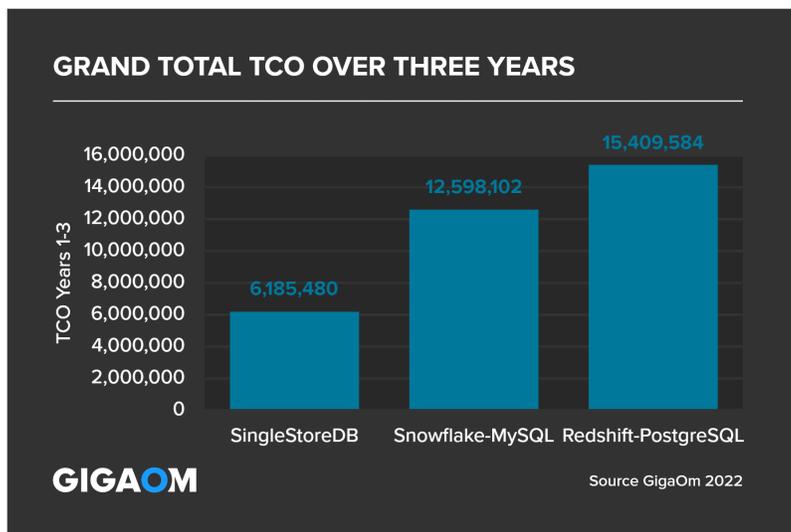


Figure 8. Grand Total TCO Over Three Years (lower is better)

6. Conclusion

Organizations run their day-to-day businesses with transactional applications and databases, while they glean insights and make critical decisions using analytical databases and business intelligence tools. Transactional workloads are relegated to database engines designed and tuned for transactional high throughput. Meanwhile, the big data generated by all the transactions requires analytics platforms to load, store, and analyze volumes of data at high speed, providing timely insights to businesses.

This calls for two different database architectures and platforms: online transactional processing (OLTP) platforms to handle transactional workloads and online analytical processing (OLAP) engines to perform analytics and reporting.

Today, with the evolution of modern interactive SaaS applications, there is a growing focus on operational analytics that include streaming data ingest and analysis on operational data in real time. Some refer to operational analytics as hybrid transaction/analytical processing (HTAP), translytical, or hybrid operational analytic processing (HOAP). Organizations are often reluctant to attempt analyzing real-time data, fearing the analytical workload will hamper the performance of the operational work that has to be the priority.

There is one exception available on the market—SingleStoreDB. Our benchmark results were insightful in revealing SingleStoreDB's competitiveness in operating effectively, actually putting it in a winning position for both transactional and operational workloads. The use of a single database facilitates operational analytics and offers an efficient approach for any organization.

For the TPC H-like workload, SingleStoreDB obtained a 21.5 geometric mean for both the “no refresh” and “with refresh” workloads, which was better than both of the pure-play data warehouses—Snowflake and Redshift. Snowflake outperformed Redshift in the TPC H-like workload.

In the TPC DS-like workload, Redshift was superior, both with maintenance and without maintenance. Its 4.1 geometric mean outperformed SingleStoreDB's 6.7 without maintenance, while its 3.9 with maintenance likewise bested SingleStoreDB's 7.0. In both tests, SingleStoreDB outperformed Snowflake in both metrics.

Given the vast superiority in transactional processing and the high competitiveness in analytic processing, the efficiencies of one database—SingleStoreDB—across the spectrum of enterprise

needs should be considered.

Platform costs favor SingleStoreDB by 1.9x over Snowflake-MySQL and 2.5x over Redshift-PostgreSQL in Year 1, but the time-effort costs are where even greater savings is found. Time-Effort costs favor SingleStoreDB by 2.7x over Snowflake-MySQL and 2.3x over Redshift-PostgreSQL in Year 1.

We calculated the annual costs of the platform stacks and the Time-Effort Costs (People Costs, Development Costs and Production Costs) and concluded that **SingleStoreDB is 2 times cheaper than the Snowflake-MySQL stack** and **2.5 times cheaper than the Redshift-PostgreSQL stack** over 3 years running enterprise-equivalent workloads.

7. Disclaimer

Cost is important but it is only one criterion for a database platform selection. This test is a point-in-time check into specific costs. There are numerous other factors to consider in selection across factors of Performance, Administration, Features and Functionality, Workload Management, User Interface, Scalability, Vendor, Reliability, and numerous other criteria. It is also our experience that costs change over time and are competitively different for different workloads. Also, a cost leader can hit up against the point of diminishing returns and viable contenders can quickly close the gap.

GigaOm runs all of its tests to strict ethical standards. The results of the report are the objective results of the application of tests to the simulations described in the report. The report clearly defines the selected criteria and process used to establish the field test. The report also clearly states the tools and workloads used. The reader is left to determine for themselves how to qualify the information for their individual needs. The report does not make any claim regarding the third-party certification and presents the objective results received from the application of the process to the criteria as described in the report. The report strictly measures TCO and does not purport to evaluate other factors that potential customers may find relevant when making a purchase decision.

This is a sponsored report. SingleStore chose the competitors, the test, and the SingleStoreDB configuration was recommended by SingleStore. GigaOm chose the most compatible configurations for the other platforms listed. Choosing compatible configurations is subject to judgment. We have attempted to describe our decisions in this paper.

8. About SingleStore

We live in a changing world shaped by data. Data is pervasive in all aspects of our lives—in various shapes, speeds, sizes and times. Businesses use this data in its various formats to create limitless experiences, something consumers and customers expect to be readily available at their fingertips. And, this data must be processed faster, no matter how complex, large or frequently it occurs. Today, businesses find themselves in the data-intensive era.

Several trends—including faster internet, mass adoption of modern applications, and the need for unified transactional and analytical processing — drove database market disruption to lead us to the present, data-intensive era. The world's leading brands rely on data—whether they're preventing millions of fraudulent credit card transactions, offering dynamic pricing to hundreds of thousands of riders, or sending title recommendations to millions of viewers. This access to real-time, actionable insights empowers them not only to make the right business decisions, but to deliver extraordinary customer experiences—and a powerful engine that unifies and simplifies your data is the most critical enabler of that success.

SingleStore delivers the world's fastest distributed SQL database for data-intensive applications and real-time analytics, SingleStoreDB. By combining transactional and analytical workloads in a single engine, SingleStore eliminates performance bottlenecks and unnecessary data movement to support constantly growing, demanding workloads.

SingleStore is designed for the world's SaaS builders, data leaders, and digital disruptors. Brands like Hulu, Uber, Comcast, and more choose SingleStore to reduce operational and design burdens of their data-intensive applications to supercharge limitless data experiences. That means riders quickly connect with their drivers. Investment and lending decisions are made within seconds. Students get learning materials as soon as they need it. Patients are treated faster, and with care.

With SingleStore, brands use their data to power the world.

9. About William McKnight

William McKnight is a former Fortune 50 technology executive and database engineer. An Ernst & Young Entrepreneur of the Year finalist and frequent best practices judge, he helps enterprise clients with action plans, architectures, strategies, and technology tools to manage information.

Currently, William is an analyst for GigaOm Research who takes corporate information and turns it into a bottom-line-enhancing asset. He has worked with Dong Energy, France Telecom, Pfizer, Samba Bank, ScotiaBank, Teva Pharmaceuticals, and Verizon, among many others. William focuses on delivering business value and solving business problems utilizing proven approaches in information management.

10. About Jake Dolezal

Jake Dolezal is a contributing analyst at GigaOm. He has two decades of experience in the information management field, with expertise in analytics, data warehousing, master data management, data governance, business intelligence, statistics, data modeling and integration, and visualization. Jake has solved technical problems across a broad range of industries, including healthcare, education, government, manufacturing, engineering, hospitality, and restaurants. He has a doctorate in information management from Syracuse University.

11. About GigaOm

GigaOm provides technical, operational, and business advice for IT's strategic digital enterprise and business initiatives. Enterprise business leaders, CIOs, and technology organizations partner with GigaOm for practical, actionable, strategic, and visionary advice for modernizing and transforming their business. GigaOm's advice empowers enterprises to successfully compete in an increasingly complicated business atmosphere that requires a solid understanding of constantly changing customer demands.

GigaOm works directly with enterprises both inside and outside of the IT organization to apply proven research and methodologies designed to avoid pitfalls and roadblocks while balancing risk and innovation. Research methodologies include but are not limited to adoption and benchmarking surveys, use cases, interviews, ROI/TCO, market landscapes, strategic trends, and technical benchmarks. Our analysts possess 20+ years of experience advising a spectrum of clients from early adopters to mainstream enterprises.

GigaOm's perspective is that of the unbiased enterprise practitioner. Through this perspective, GigaOm connects with engaged and loyal subscribers on a deep and meaningful level.

12. Copyright

© [Knowingly, Inc.](#) 2022 "*Transactional and Analytical Workloads*" is a trademark of [Knowingly, Inc.](#).
For permission to reproduce this report, please contact sales@gigaom.com.