A Look Inside the SQream Data Analytics Platform

WHITEPAPER

sqream.com

Bringing Data Analytics into the Modern Age

In the past, data was small, as were the number of data consumers. Most datasets were relatively simple, coming from a handful of ERP, CRM and other transactional sources. Traditional data analytics solutions were built to support this type of data. As computing hardware advanced, these databases got faster 'for free.' However, they have by now become legacy technology incapable of utilizing new parallelized computing paradigms.

Today, data drives business. Every website we visit, connected device we use, or TV show we watch generates data that impacts the businesses behind them. Data yields insights that help organizations make good business decisions and stay competitive. But with the gap between the level of data their systems were built to handle, and the massive volumes they face today, businesses are finding their data analytics solutions to be a big problem.

Ad-hoc Querying Is King

In the past, organizations used data warehouses to carry out bulk periodic reports that would be updated once a day at most, due to lengthy processes. Today's organizations have data analysts and data scientists that need "human real-time" responses to data exploration and experimentations. Empowering business-critical employees and their tools and frameworks means that immediate, unrestricted access is now the gold standard.

Modern Technology Presents Amazing Opportunities

Modern technological advancements like cutting-edge high-throughput hardware acceleration and flexible cloud infrastructure present amazing opportunities for modernizing the way businesses access their data. Legacy solutions weren't designed to take advantage of high throughput compute like multicore processors and hardware-accelerated algorithms.

Access To More Data Is A Necessity

To get around the classic data warehouse limitations, many enterprises have implemented unstructured data lakes, often built around Hadoop data stores, that promise an alternative to data processing.

These data lakes serve to store large amounts of semi-structured and unstructured data. Their flexibility led to the idea of "schema-on-read," which in turn has led to the increasing need for intense data preparation. The need to prepare every piece of data with uncommon skill sets has shifted autonomy away from data analysts and data scientists, who now must jump through hoops to access the data they need.

The Hadoop ecosystem has ultimately failed to deliver the flexible, interactive access to data that it promised, and has discouraged data professionals by hiding data behind programming APIs and a wide slew of difficult-to-use, inadequate tools.

SQream: A GPU-Accelerated Data Analytics Solution

When founding SQream, we looked at data warehouses and realized that hardware-accelerated coprocessors can be a key component in making more data more accessible. While hardware-accelerated coprocessors aren't new, they were previously based on custom FPGAs and exotic hardware that was expensive to buy and maintain. The recent popularity of machine learning, AI, and even cryptocurrency has brought the GPU coprocessor to all hardware vendors and cloud infrastructures, making them more powerful and more accessible than ever before.

GPUs are many-core accelerator cards typically designed for graphics processing. Their power comes not only from their large number of compute cores, but also from incredible memory bandwidth, along with the software development tactics used to develop for them. GPUs allow software developers to parallelize complex tasks, and do so with performance that's difficult to achieve on classic CPU-bound implementations. For example, compression, encryption, and sorting algorithms benefit from the GPU's high core count and memory bandwidth.

As a result, GPUs allow for much better resource utilization, with the opportunity to scale up and out with additional GPUs when necessary. A handful of GPU-enabled servers can support a large enterprise's data compute needs, while delivering faster performance at a fraction of the cost of competing CPU-only solutions.

Built Ground-up, from Scratch

SQream, founded to address the growing frustration with existing data warehousing systems, has created the first enterprise-grade GPU-accelerated data analytics platform.

Rather than building on unsuitable technology stacks like Hadoop or Postgres, SQream was created from scratch to empower data consumers. It was built to harness the raw brute-force power and high throughput capabilities of the GPU, with MPP-on-chip capabilities and a fully relational SQL database.

SQream is not an in-memory database, or an SQL translation layer for Hadoop. It is its own database, designed for larger-than-memory, constantly growing data. SQream provides the same solution for on-premise and cloud environments, enabling easy implementation of hybrid and migration solutions on both.

On-prem and on the cloud, SQream accelerates ETL/ELT and data preparation processes, and provides faster analytic (query) results – TTTI (Total Time to Insight). TTTI reflects the time it takes to generate an effective insight from data creation. The idea is that analytics provides value only once you have insight. Focusing only on query time misrepresents the actual time it takes to produce analytics, as time spent on ingesting large quantities of data is also critical in achieving time-sensitive data insights..

Throughout the process, from data preparation to ingestion and insights, SQream provides the best cost performance solution available, on-prem and on the cloud.

SQream for the Cloud

SQream for the cloud is an auto-provisioning, self-onboarding, scalable, native-cloud analytics service. Immediately upon signing, the user is provisioned with a fully scalable DB service geared towards achieving rapid Total Time To Insight (TTTI).

Architecturally, the SQream cloud offering is a self-contained, containerized solution and can be deployed as several options:

- Run as a seamless provisioning service. The service is run and managed by SQream, using SQream's cloud account.
- Private cloud on a customer's public cloud account i.e.: AWS, GCP, Azure, etc. Still seamless operation and seamless scaling.
- Private cloud on customer infrastructure.

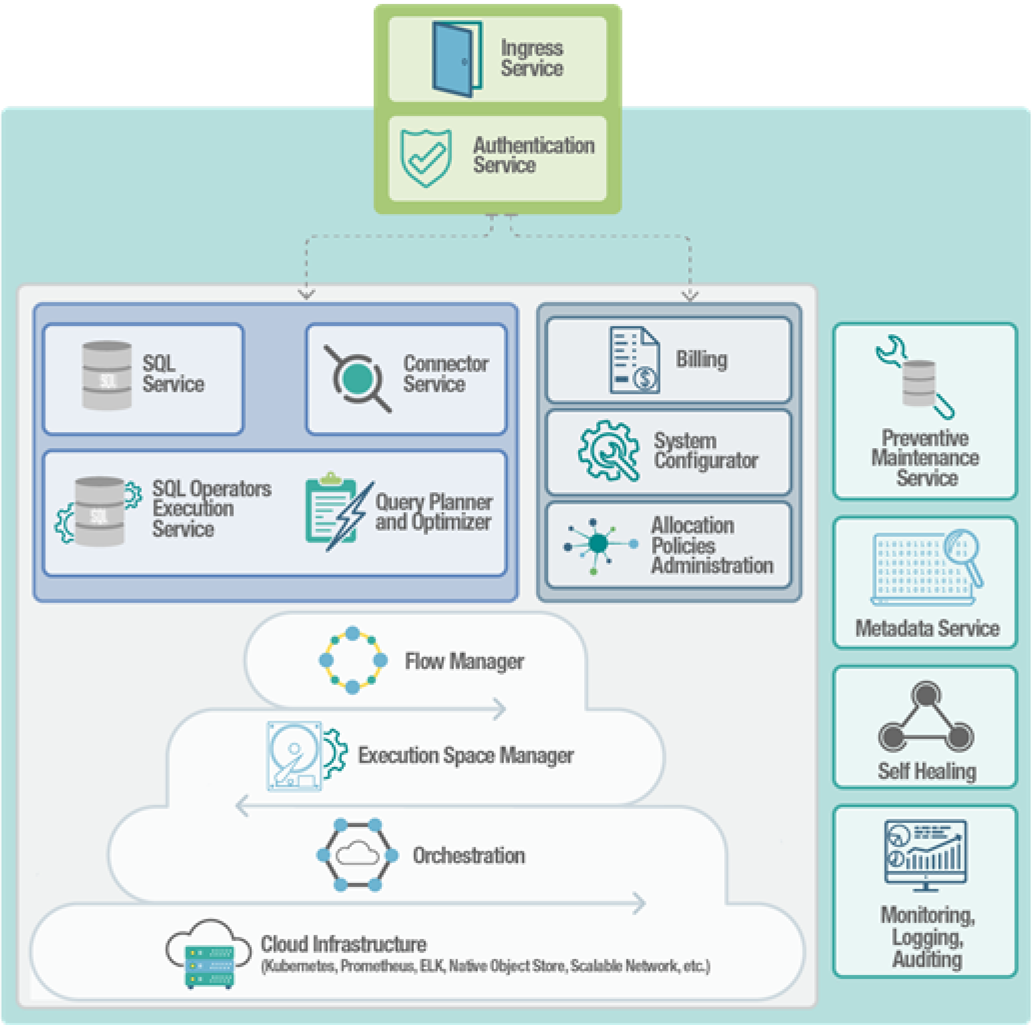

Ingress Service
Authentication Service
SQL Service
Connector Service
SQL Operators Execution Service
Query Planner and Optimizer
Billing
System Configurator
Allocation Policies Administration
Preventive Maintenance Service
Metadata Service
Self Healing
Monitoring, Logging, Auditing
Flow Manager
Execution Space Manager
Orchestration
Cloud Infrastructure
(Kubernetes, Prometheus, ELK, Native Object Store, Scalable Network, etc.)

Figure 1 - SQream for the Cloud

SQream - Core Concepts

1. GPU Utilization

SQream uses the GPU to achieve parallel data processing. By splitting large tasks into smaller processes, SQream distributes operations between multiple GPU cores. GPU-accelerated architecture and automated optimizations remove intermediate steps when analyzing data.

SQream utilizes the GPU's brute power to analyze data immediately after loading. It achieves this by compressing and collecting metadata while data is loaded, resulting in reduced I/O and higher throughput for loads and queries.

2. Separation Between Compute, Storage and Metadata

The hardware elements. Even today's modern NoSQL data traditional data warehouses tightly couples elements, and sometimes include well tightly couple storage and compute together on the same infrastructure nodes. Conversely, SQream completely separates between compute and storage, running multiple compute units to store or retrieve data from a single or multiple storage sources. This concept provides flexibility and easy scaling, allowing SQream compute to be used with an existing storage solution.

The separation of metadata from storage and compute allows each compute node to read/write without the need to slow down and coordinate with other nodes.

This means there is no diminishing return for additional machines and thus SQream achieves linear scalability and no throughput limit.

SQream's method of decoupling compute, storage and other resources is described in the architectural diagram below:

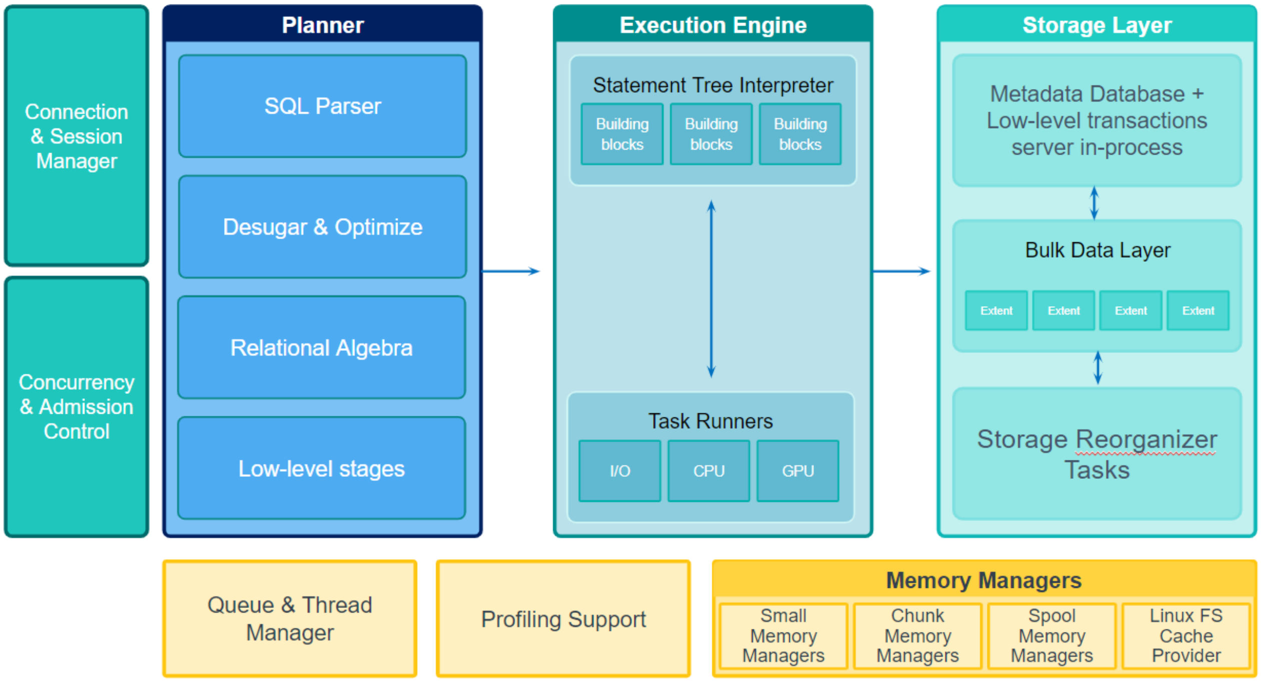


Figure 2 - High Level Architectural Diagram of SQream Internals

As shown above, SQream's system architecture physically separates the planner, runtime and storage layers, and performs communication via message passing.

3. Columnar DB

A columnar DB stores data tables by column rather than by row. Benefits include more efficient access to data when only querying a subset of columns (by eliminating the need to read columns that are not relevant), and more options for data compression. As such, it's perfect for running online analytical processing (OLAP) storing metadata and efficient for real-time analytics. The main reason that columnar DB is efficient for these tasks is because it excels at loading new data quickly.

4. Chunking

Chunking refers to strategies for improving performance by using special knowledge of a situation to aggregate related memory-allocation requests. Each chunk contains multiple values of the same column. This speeds up the GPU and parallel processing. The acceleration is done at a few levels:

1. Saving the same data type in chunks, enables better parallelism when processing the chunk in a multiple core GPU (example - take a value, add +5. Transformation will be done in parallel depending on the GPU core amount).
2. Allowing each compute node to access (read/write) each chunk, allows each node to read/write at its maximum speed, without slowing down to coordinate i/o with other nodes. The settlement of ACID (atomicity, consistency, isolation, durability) integrity is done downstream by the metadata service. Effectively, having 100 nodes will result in 100x throughput.

Using Chunking, the following capabilities are enabled:

Partition Data in Multiple Dimensions –

- Vertical partitioning – columnar engine – This feature allows selective access to the required subset of columns, reducing disk scan and memory I/O when compared with standard row storage. This concept is well-suited for parallelized compute, like the GPU.
- Horizontal partitioning – chunks and extents – SQream automatically splits up the storage horizontally into manageable chunks enabling efficient use of the hardware resources and relatively small GRAM (GPU RAM) availability in GPUs. The clever use of spooling and caching help make the most of the limited GRAM.
- No bottlenecks sync when reading/writing data – the data structure is consolidated post-write by the metadata server so there is no diminishing throughput when adding more readers. Each reader has the same speed access to any chunk/extent, so there is no slow down in reading (as opposed to data affinity architecture).

•**Automatically Performed During Data Ingestions** – SQream features automatic vertical and horizontal partitioning that is performed on-the-fly without any user intervention, as shown in the following figure:

•**Enables Scaling to Petabytes** – SQream handles data ranging in size from standard terabytes to increasingly relevant petabytes, minimizing complex scaling processes in minimum time. (See our benchmark chapter).

5. Metadata

While storing data in chunks, SQream also updates metadata, used for indicating the location and context of each chunk. Metadata and compressed chunks are stored separately.

Metadata indicates where specific data is stored and enables SQream to identify and skip unnecessary query data. This reduces the overall I/O across the disk, network, RAM, PCIe, and GPU RAM interfaces.

Metadata Usage Benefits

Using metadata provides the following benefits:

- **Linear scaling with no diminishing throughput when increasing reader/writer processes** – Each writer doesn't have to sync with others when writing. There is no sync before or during the actual write. Only at the end of a transaction, all affected extends are updated to the metadata so there is no slow down when adding more machines, just increased throughput.
- **Collection** – Automated and transparent data collection across all data types and columns, requiring no manual intervention or maintenance.
- **Storage efficiency** – Space-efficient compared to columns, resulting in less than 1% overhead.
- **Data skipping** – Accelerated querying, as calculated zone-maps enable efficient data skipping (also known as data pruning) to eliminate reading irrelevant data.

6. Compressed Data

SQream default compression mode is auto-compression, in which the system tries to determine the best compression algorithm based on the actual data.

Automatic Adaptive Compression – Optimized for Query Performance

SQream compression can be done on-the-fly with limited/no impact on i/o speed. In addition, each chunk contains the same data type internally, so it usually lends itself to a specific compression method. SQream 'test drives' which compression method is best for each column/chunk and find the most cost effective one.

1:5 Compression means saving 80% on storage & I/O

SQream is able to compress large datasets by 80%. That means that reading actual data is 5 times faster than the network speed and the actual storage requirements are 5 times lower than the uncompressed data size.

SQream Features and Interfaces

Fully Featured SQL and Industry-standard Connectivity

SQream supports ANSI-92 SQL compliant syntax. It easily integrates into existing ecosystems, with support for industry standard ODBC and JDBC connectors, as well as Python and C# .Net, C++, Java, and others.

SQream's native SQL interface eases transition from other databases. There's no need to maintain odd APIs and custom Scala code. Full SQL support lets any existing ETL and applications connect and offload heavy database operations to SQream, minimizing the time needed to get up and running with the new platform.

More information about SQL support can be found in the SQream SQL Reference.

SQream adaptors are geared for high throughput employing techniques such as connection pooling and multiplexing. This works effectively both with highly chatty connections, as well as high throughput ones.

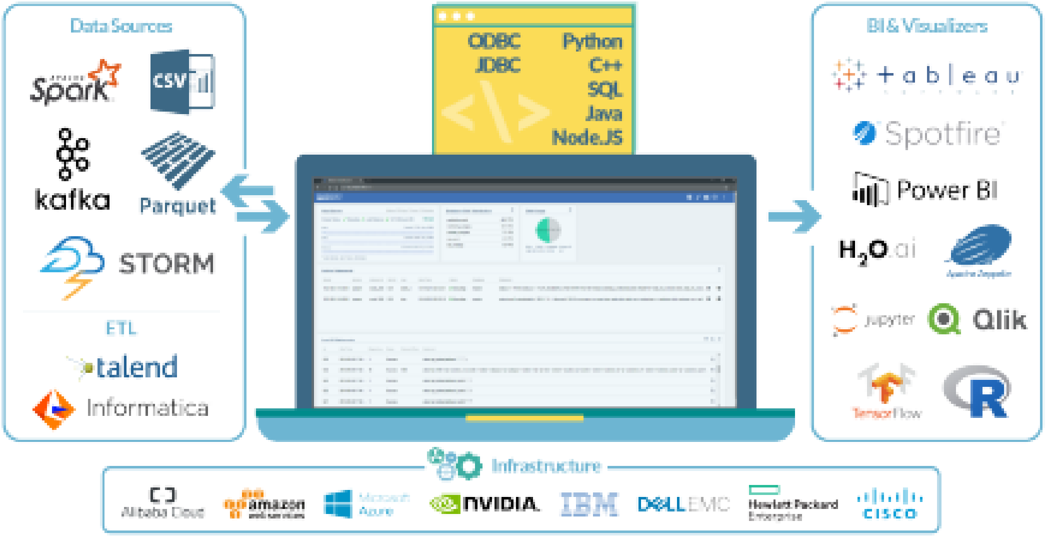


Figure 3 - SQream can be deployed on any hardware, and connected to all BI tools with JDBC and ODBC support

Automatic Tuning, Self Management

SQream's interface layer contains hundreds of optimizations and automations designed to let businesses focus on data, rather than data management.

Most databases require a team of administrators to finesse and manually tune processes, maintain indexing, update views and projections, etc. SQream was designed for frequently-changing, modern workloads. It was built to handle worst-case scenarios, and is optimized for huge datasets, where typical database optimizations struggle.

SQream's transparent metadata collection and adaptive automatic compression let data consumers run queries on hundreds of terabytes of data, where other databases simply can't function (try indexing a 500 TB dataset!).

Analyze Raw Data Directly and Easily

SQream's automatic tuning is a key enabler for analyzing data without intermediate steps. The raw, brute power of the GPU allows SQream to analyze data immediately after load. This is in stark contrast to most data warehouses, which require time-consuming and insight-limiting processes like indexing, cubing, projecting, etc.

SQream parallelizes all aspects of ingest: reading, processing and writing steps, with no user intervention. During the ingest process, SQream automatically and transparently prepares all data for immediate, fast analysis – with no user intervention required.

Query from Foreign Tables

Foreign tables can be used to run queries directly on data without first inserting it into SQream. The platform supports read only foreign tables, so you can query from foreign tables, but you cannot insert to them, or run deletes or updates on them. Although foreign tables can be used without inserting data into SQream, one of their main use cases is to help with the insertion process. An insert select statement on an foreign table can be used to insert data into SQream using the full power of the query engine to perform data manipulation required during load.

From ETL to ELT

ETL is a continuous, ongoing process with a well-defined workflow: ETL first extracts data from homogeneous or heterogeneous data sources. Next, it deposits the data into a staging area. From there, the data goes through a cleansing process, gets enriched and transformed, and is finally stored in a data warehouse. ETL requires detailed planning, supervision and coding by data engineers and developers. Even after designing the process, it takes time for the data to go through each stage when updating the data warehouse with new information.

ELT stands for "Extract, Load, and Transform." In this process, data gets leveraged via a data warehouse in order to do basic transformations. ELT uses cloud-based data warehousing solutions for different types of data – including structured, unstructured, semi-structured, and even raw data types. The ELT process also works hand-in-hand with data lakes. Data transformation is still necessary before analyzing the data with a business intelligence platform. However, data cleansing, enrichment and transformation occur after loading the data into the data lake. ELT is usually used in scalable cloud infrastructure which supports structured and unstructured data sources.

The SQream platform supports both ETL and ELT. You can leverage SQream capabilities on Cloud and on Prem.

ETLs are usually part of a data pipeline. Data pipelines usually deliver data from multiple sources to their target audience, be it another system or another data platform. The pipelines are used to deliver the right data, for the right constituency, in just the way it needs to be consumed. This is usually achieved by a series of interactive ELTs, each feeding into one or more iterations.

The overall effectiveness of multiple ETLs is determined by two factors:

- Timeliness: is the ETL able to deliver the data through the pipeline in time so the consuming resource is able to utilize and create value in time? If we delay the consumption of insight downstream, we lose time (and money). So a speedy process is critical when a downstream platform requires timely insight.
- Quality of time sensitive data. In cases where the data is time sensitive, there is a window of opportunity after which there is a diminishing return with the passage of time. If we deliver insight on stale data, it's worthless.

In each ETL process, the data is delivered and some preparation is performed, so the downstream activities can run faster and more correctly. This means that there is a need for:

- Ingesting the data rapidly (the 'load' part).
- Processing the data rapidly (the 'transform' part).
- Doing some processing to accelerate the upstream process.

SQream excels at ingestion: The core processing by GPU is both extremely fast and cost-effective. The separation of storage and compute, coupled with a unique approach of 'effective sharing' (as opposed to strictly 'share everything' or 'share nothing') enables each GPU to ingest and process without a need to coordinate with others, thus eliminating bottlenecks.

SQream scales across multiple machines and GPUs, and as such, has lots of extra bandwidth. This is used to perform processing, on the fly, to accelerate other activities without slowing down the process, ie: sorting, compressing/decompressing and filtering.

SQream loads faster, does the processing faster and while doing so, also executes preparations for downstream acceleration – all at a very cost-effective rate.

When users need data from multiple sources delivered to multiple constituencies, often transformed and blended, and it's data at scale, they face a choice: either retrieve up-to-date data from huge data sets and risk very slow response time and potentially high cost, or build a complex data pipeline that will deliver the right data to the right user, ready to be consumed, but not up-to-date.

When the data or the process has time sensitivity, the data currency (how up to date is it?) is a significant factor.

There are multiple techniques for data pipeline (ETL/ELT, streaming, or lambda architecture). All share the inherent complexity and time delay of a data pipeline.

SQream allows the user a blended approach that enables timely delivery of up-to-date data, ready to be consumed by the target system / user. SQream can query effectively huge data sets, as well as accelerate complex data pipeline by fast ingest and fast processing.

Integration and Fast Data Ingest

One of the most common tasks for any analytics database is loading data from an external source. SQream ingests up to 3.5 TB per hour per GPU from a variety of sources, either directly from flat files like CSV or Parquet, or through a variety of industry accepted ETL tools.

SQream can also read data directly from external sources using the foreign table syntax, which avoids loading data before it is needed.

Streaming Analytics vs Batch Analytics

Kafka Integration

A very common tool framework is Apache Kafka, which uses a publish and subscribe model to write and read streams of records. Kafka Connect is a framework for connecting Kafka with external systems, including databases like SQream.

SQream implemented the Kafka integration capabilities for various Kafka components to allow continuous data flow from and to our work environment, and to allow the full scope of Kafka added value.

It is common for SQream to provide the analytics database, where Apache Kafka serves as the messaging queue system, and Apache Spark provides transformations. In such installations, SQream will be the layer bridging the applications with persistence stores for analysis.

File Format Support

In addition to network-based connectivity, SQream also supports native reading and writing of the following popular file formats:

- **CSV** – A comma-separated values (CSV) file is a delimited text file that uses a comma to separate values. Each line of the file is a data record.
- **Apache Parquet** – Parquet is built to support very efficient compression and encoding schemes. Parquet allows compression schemes to be specified on a per-column level, and is future-proofed to allow adding more encodings as they are invented and implemented.
- **Apache ORC** – The Optimized Row Columnar (ORC) file format provides a highly efficient way to store Hive data. It was designed to overcome limitations of the other Hive file formats. Using ORC files improves performance when Hive is reading, writing, and processing data.
- **JSON** – JavaScript Object Notation is a lightweight data-interchange format. SQream supports JSON files consisting of either a continuous batch of JSON objects, or an array of JSON objects.
- **Apache Avro** – Avro is a well-known data serialization system that relies on schemas. Due to its flexibility and nesting as an efficient data storage method, SQream supports the Avro binary data format as an alternative to JSON. SQream supports Primitive Data Types; Complex Data Types and logical data types.

By connecting to a wide variety of file formats, SQream allows users to integrate easily with the various existing environments used across the organization.

Object-level Roles and Permission System

SQream offers an object-level permission system, with roles and object control all the way down to per-table authentication. More information about this feature can be found in the SQream SQL Reference.

Direct IT Monitoring

SQream runs on standard hardware and Linux distributions like CentOS, RedHat, and Ubuntu. This means you can easily integrate with any control and monitoring software you use to track your Linux-based machines. SQream is routinely integrated with enterprise and open source solutions.

Extensive Logging

SQream contains a built-in logger that tracks critical server information, enabling your IT and security teams to gain insights into the server's operation from failed login attempts to GPU / CPU time spent per query, and read-write cycles to memory.

User Authentication

SQream supports two modes of user authentication:

1. Database Authentication: user-password authentication is done against passwords saved in SQream's metadata and layered in a secure way.

2. LDAP or alternatively, Federated Authentication: user-password authentication is done against an external (i.e. "federated") source, for example the customer's own active directory services.

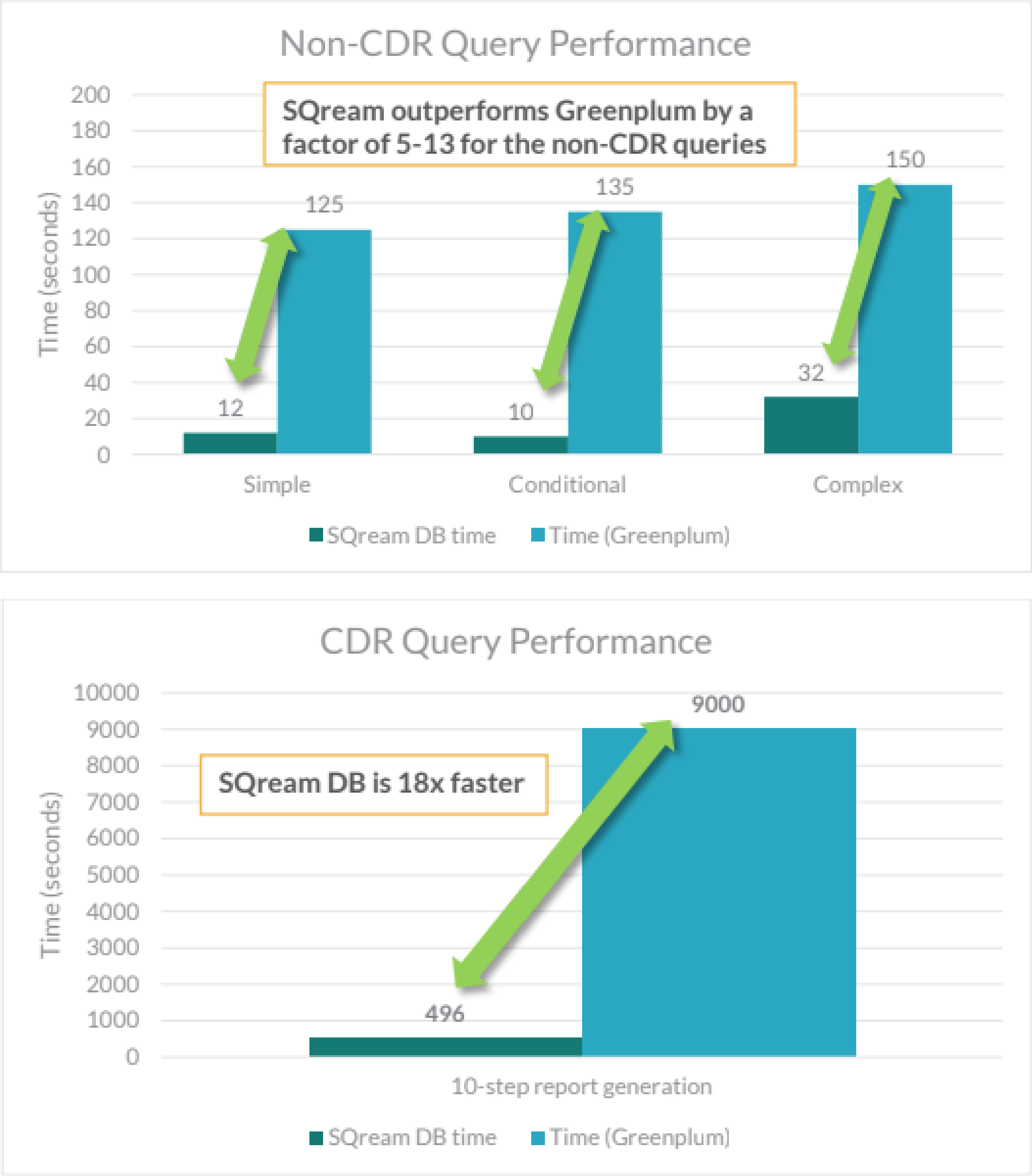
GDPR

SQream helps organizations reduce the burden of GDPR compliance. Although it is the responsibility of the customer to protect the personal data of their end users / consumers, SQream provides them with the means to do so inside the SQream ecosystem. This includes access control and account management with a strong password policy; 2-factor authentication and credential storage and encryption; data encryption; and an audit trail for tracing potentially compromised data.

Other GDPR elements supported by SQream include data retention for a select period of time, privacy notices through UI and CLI, and data transfer via connectors.

SQream also offers an asset registry, which provides a repository that records assets, systems and applications used for processing or storing personal data across the organization.

Query types and runtimes

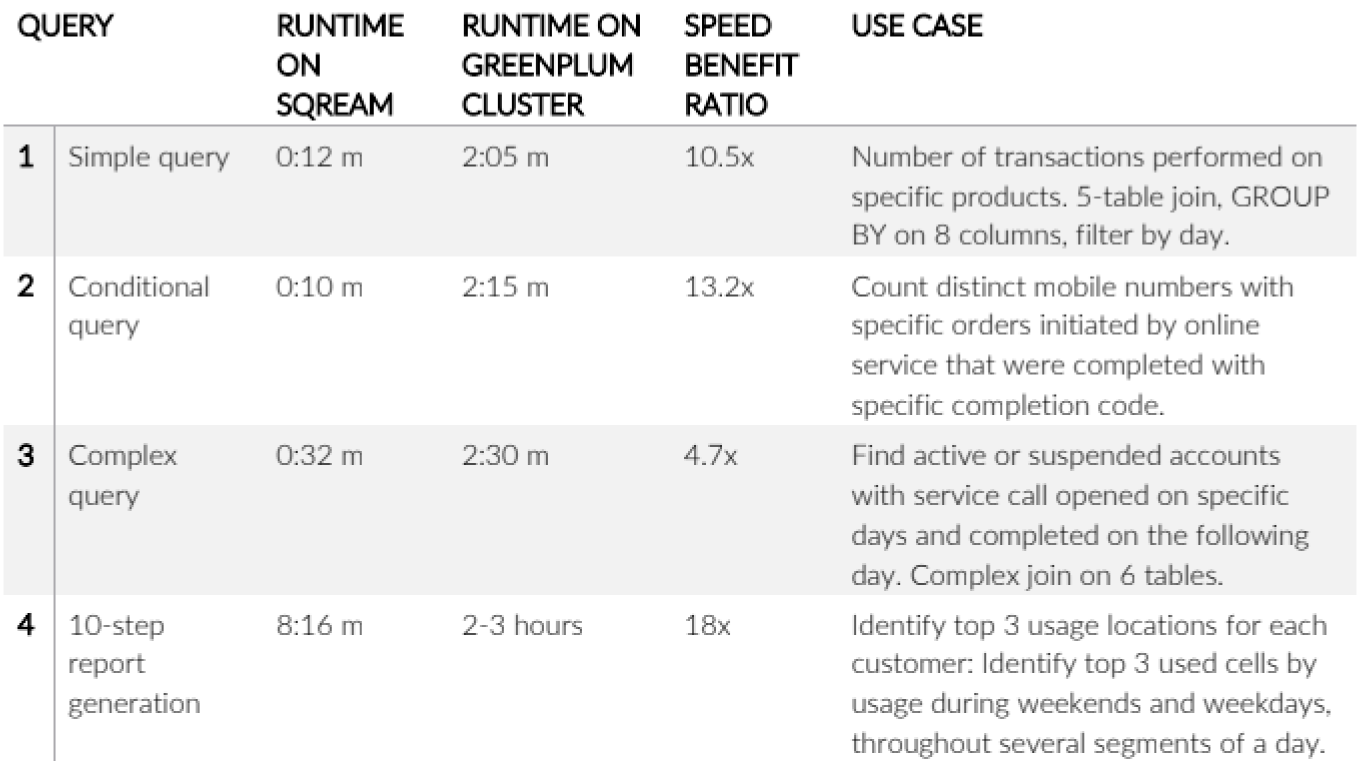

Non-CDR Query Performance

SQream outperforms Greenplum by a factor of 5-13 for the non-CDR queries

CDR Query Performance

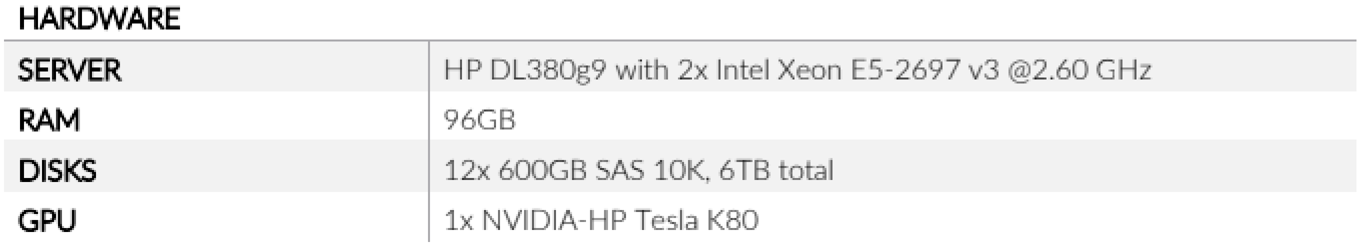
SQream DB is 18x faster

Figure 4 - CDR and non-CDR query performance, SQream versus Greenplum



QUERY		RUNTIME ON SQREAM	RUNTIME ON GREENPLUM CLUSTER	SPEED BENEFIT RATIO	USE CASE
1	Simple query	0:12 m	2:05 m	10.5x	Number of transactions performed on specific products. 5-table join, GROUP BY on 8 columns, filter by day.
2	Conditional query	0:10 m	2:15 m	13.2x	Count distinct mobile numbers with specific orders initiated by online service that were completed with specific completion code.
3	Complex query	0:32 m	2:30 m	4.7x	Find active or suspended accounts with service call opened on specific days and completed on the following day. Complex join on 6 tables.
4	10-step report generation	8:16 m	2-3 hours	18x	Identify top 3 usage locations for each customer: Identify top 3 used cells by usage during weekends and weekdays, throughout several segments of a day.

Server Configuration



HARDWARE	
SERVER	HP DL380g9 with 2x Intel Xeon E5-2697 v3 @2.60 GHz
RAM	96GB
DISKS	12x 600GB SAS 10K, 6TB total
GPU	1x NVIDIA-HP Tesla K80

Summary

In today's database market, SQream offers significantly better cost-performance than other market players, specifically in the multi-terabyte range where scaling with CPUs is not cost-effective. With standardized SQL, superior scaling and a robust architecture based on standard hardware, SQream is a future-proof big data solution.

SQream brings the opportunity to do more with more data. Fast insights with hundreds of billions of data points are now within reach. SQream can be integrated as a standalone database solution, or as a complementary analytics database, maximizing your IT investments.

The integration of SQream is an easy transition from other SQL databases. There is little-to-no rewriting of SQL queries. SQream connects easily to your existing ecosystem.

Because SQream uses standard SQL and common language bindings, deep learning technologies that also use GPUs, such as TensorFlow and Theano, work 'hand in glove' to reduce the time for modeling and learning experiments.

SQream enables data scientists to be more productive, so that they can perform many more variations of the parameters of a model in the same time periods as it would normally take to do a few simple variations – and on much less hardware.

About SQream

SQream provides an analytics platform that minimizes Total Time to Insights (TTTI) for time-sensitive data at-scale, both on-prem and on-cloud. Designed for the new category of tera-to-peta-scale data, the GPU-powered platform enables enterprises to rapidly ingest and analyze their growing data – providing full-picture visibility for improved customer experience, operational efficiency, increased revenue, and previously unobtainable business insights.

SQream is trusted by leading enterprises including LG Uplus, Pubmatic, ACL, AIS, and more. To learn more, visit sqream.com or follow us on twitter @sqreamtech.

Bring the power of SQream to your business –

info@sqream.com | sqream.com | @SQreamtech