SQREAM

A LOOK INSIDE SQREAM DB

ACCLERATED DATA WAREHOUSE FOR ANALYTICS

WHITE PAPER

BRINGING DATA WAREHOUSING 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 warehouses 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, and 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 warehouses to be a big problem.

AD-HOC QUERYING IS KING

In the past, data warehouses were used for 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 skillsets 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 DB: A GPU-ACCELERATED DATA WAREHOUSE

When founding SQream, we looked at existing 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 warehouse - SQream DB.

Rather than building on unsuitable technology stacks like Hadoop or Postgres, SQream DB 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 DB 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 DB ARCHITECTURE: SHARED DATA, DISTRIBUTED COMPUTE

SQream DB physically separates storage and compute, and is built of three main components:

- **Fully-featured SQL Interface and Statement Compiler**:
  SQream DB's interface layer is a collection of services that controls the data warehouse. It includes concurrency control, access control (object-level permissions), cluster management, and a statement compiler. The modern and flexible statement compiler is written with hundreds of optimizations, micropasses and stages. It instructs the execution layer, through message passing, on which operations to run and in what order.
- **Compute and Execution layer**:
  The compute layer is where the actual data processing tasks run. This layer includes CPU and GPU resources, controlled by the SQream DB software.
- **Storage layer**:
  The storage layer is split into the metadata layer, where all routine database objects are stored, and the persistent columnar bulk data layer, which is heavily optimized for raw tablescan performance.

In traditional data warehouses, all elements are tightly coupled (sometimes even with bespoke hardware elements). Even more modern NoSQL data platforms tightly couple storage and compute together on the same infrastructure nodes. When these systems need to scale for additional concurrent users, more complex queries or even additional data storage, they suffer from performance limitations.
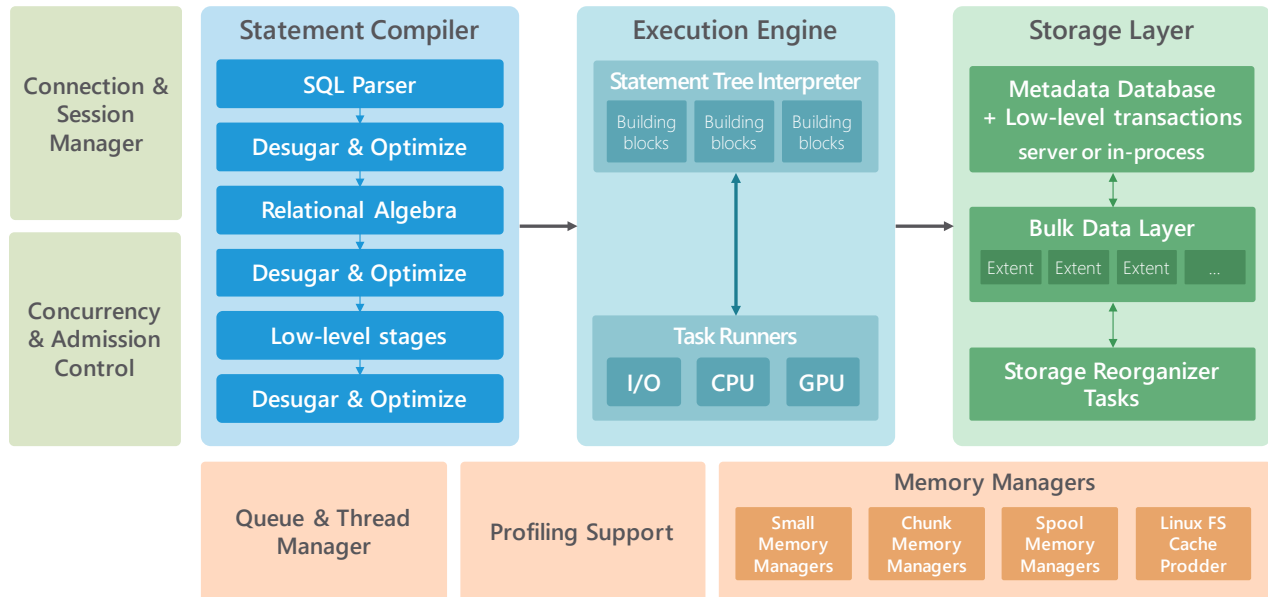

Figure 1 - SQream DB's architecture physically separates the compiler, runtime, and storage layer. Communication is performed via message passing.

AUTOMATIC HYPER-PARTITIONING DESIGNED FOR HIGH THROUGHPUT

SQream DB features two types of automatic partitioning, performed on-the-fly without any user intervention.
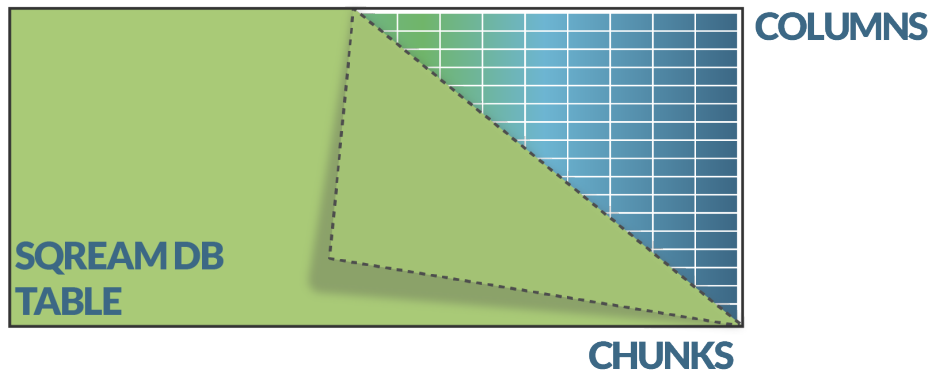

Figure 2 - The SQream DB table is partitioned vertically and horizontally

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 DB automatically splits up the storage horizontally into manageable chunks enabling efficient usage 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.

INDEPENDENT SCALING IN ANY DIRECTION

SQream DB scales each layer independently, providing businesses with the exact storage and compute resources for their dynamic needs. SQream DB's persistent storage can run on virtually any filesystem – whether local, distributed, on-premise or in the cloud - ensuring reliability and performance. SQream DB's optimized columnar storage system is partitioned both horizontally and vertically for best performance for heavy analytic operations like joins, aggregations, summarizations, and sorting.

Every SQream DB instance can be thought of as an MPP database by itself, with shared-data architecture. Each instance has full access to all data in the persistent storage layer. Permissions are managed by the service layer above the SQream DB instance, based on the user's role. These instances can be launched or shut down at will, as the requirements inevitably change.

SQream DB's architecture further relies on storage and operating system caches to transparently and automatically cache data, which can be re-used by other SQream DB instances if needed, boosting performance.
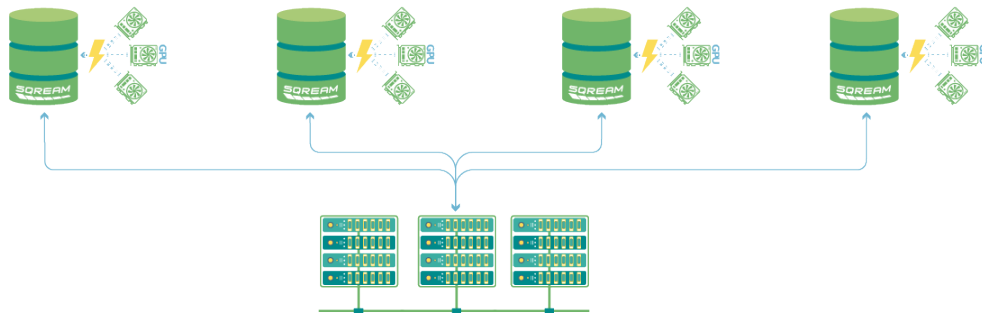

Figure 3 - SQream DB can scale both compute and storage separately, maximizing flexibility

COMBINING CPU AND GPU RESOURCES

Traditional data warehouses rely on a fixed set of resources for running all scenarios. In contrast, SQream DB can allocate more resources to handle a varied workload, by combining available CPU, GPU, RAM, and storage resources.
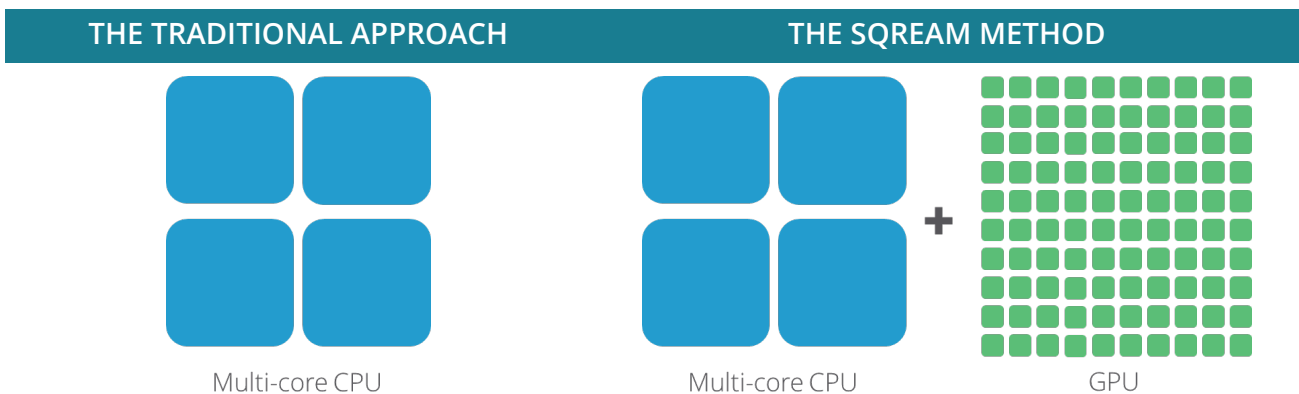

THE TRADITIONAL APPROACH	THE SQREAM METHOD
Multi-core CPU	Multi-core CPU	GPU

Figure 4 - CPU technology vs. GPU technology

This balance of CPU and GPU operations is key to ensuring optimal performance. GPUs excel at performing relatively simple, repetitive operations on large amounts of data in many streams.

For example, the SQream DB interface layer's statement compiler may decide to run some parts of the query on the CPU if it calculates that the overhead of copying data to and from the GPU would slow down the query.

FINE-GRAINED, LOW-OVERHEAD ZONE MAPS AND DATA SKIPPING

SQream DB's GPU-accelerated architecture and automatic optimizations are a key enabler for analyzing data without intermediate steps. SQream DB was developed to take advantage of the raw, brute power of the GPU, enabling data analysis immediately after loading.
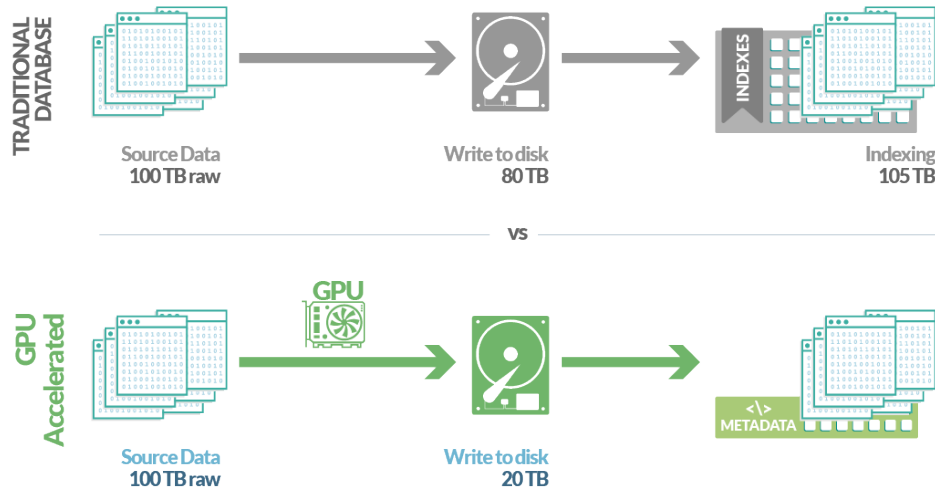

Figure 5 - SQream DB uses GPUs for compression and metadata collection during data loading, resulting in reduced I/O and higher throughput for load and queries

Upon data ingest, SQream DB collects and stores metadata for all rows stored in a chunk, as depicted in Figure 6 above. The most useful aspect of the collected metadata is the range of values and properties for the values being ingested. This metadata is stored separately from the compressed chunk.

*Figure 6 - SQream DB identifies and skips data that is unnecessary for queries,
reducing overall I/O across disk, network, RAM, PCIe, and GPU RAM interfaces*

The zone map method provides several key benefits:

- Unlike standard indexing, the metadata collected on these chunks is automatic and transparent across all data types and columns, requiring no intervention or maintenance
- The metadata collection is space-efficient compared to columns, resulting in less than 1% overhead
- Querying data becomes much faster, as the calculated zone-maps allow for efficient data pruning (also called skipping), eliminating the reading of irrelevant data
- Older data can be easily deleted

SQREAM DB FEATURES AND INTERFACES

FULLY FEATURED SQL AND INDUSTRY-STANDARD CONNECTIVITY

SQream DB supports an 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 DB'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 DB, minimizing the time needed to get up and running with the new platform.

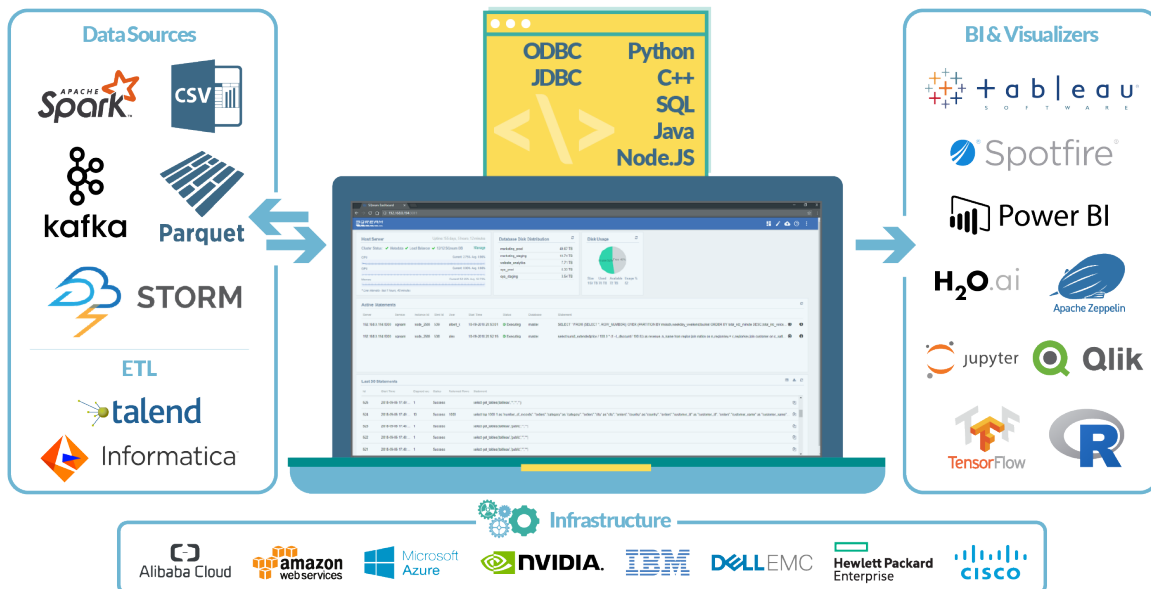More information about SQL support can be found in the SQream DB SQL Reference.

Figure 7 - SQream DB can be deployed on any hardware, and connected to all BI tools with JDBC and ODBC support

AUTOMATIC TUNING, SELF-MANAGING

SQream DB'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 DB 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 DB's automatic tuning is a key enabler for analyzing data without intermediate steps. The raw, brute power of the GPU allows SQream DB 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.

During the ingest process, SQream DB automatically and transparently prepares all data for immediate, fast analysis – with no user intervention required.

INTEGRATION AND FAST DATA INGEST

One of the most common tasks for any analytics database is loading data from an external source. SQream DB 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 DB can also read data directly from external sources using the external table syntax, which avoids loading data before it is needed.

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

OBJECT-LEVEL ROLES AND PERMISSION SYSTEM

SQream DB 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 DB SQL Reference.

DIRECT IT MONITORING

SQream DB 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 DB is routinely integrated with enterprise and open source solutions.
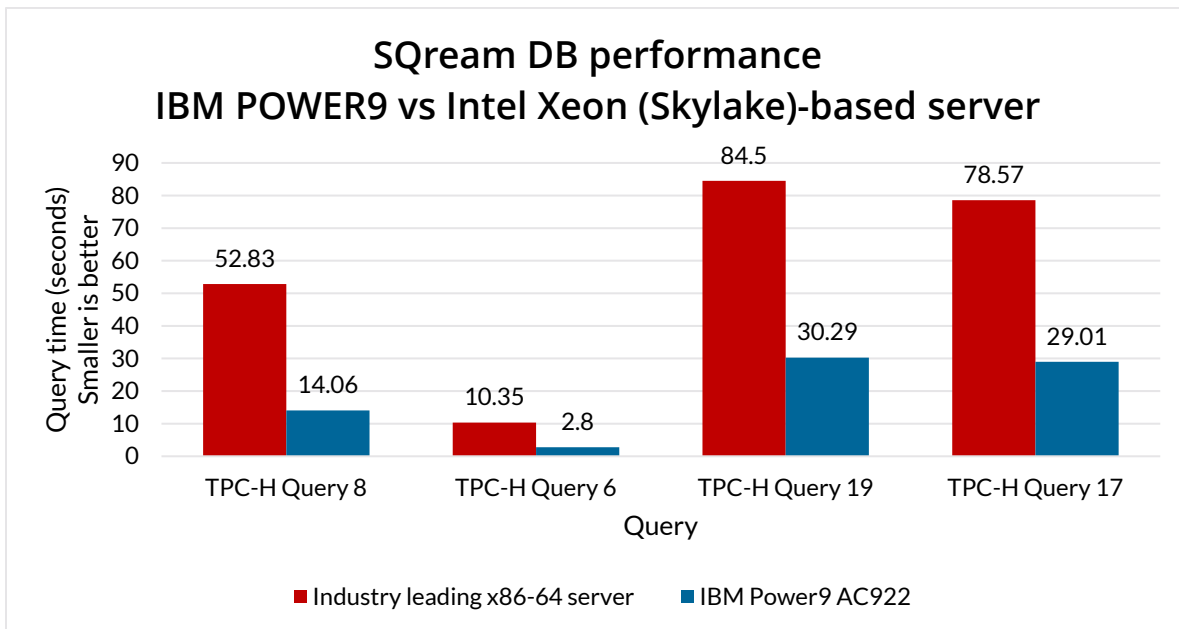
EXTENSIVE LOGGING

SQream DB 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.

FULL SUPPORT FOR IBM POWER9

IBM POWER9 is fully supported on SQream DB v3.0 and up. SQream DB on IBM Power systems is on average 150% faster than on existing x86-64 based systems.

IBM and NVIDIA's NVLink interconnect improves I/O bandwidth up to 9.5x compared to PCI Express, resulting in faster queries and more responsive dashboards. Additionally, SQream DB v3.0 introduces a wide variety of new general-purpose optimizations designed to make the most of modern compute platforms, like NVIDIA Tesla Volta on IBM POWER9. Notoriously difficult queries like multi-table joins and count distinct operations are faster than ever before, with queries up to 8.6x faster, even on hundreds of terabytes of data.

SQream DB v3.0 on POWER9 reduces heavy analytical query times by up to 3.7x in our testing, on complex benchmarking queries.

SQream DB performance
IBM POWER9 vs Intel Xeon (Skylake)-based server

Query time (seconds)
Smaller is better

Query	Industry leading x86-64 server	IBM Power9 AC922
TPC-H Query 8	52.83	14.06
TPC-H Query 6	10.35	2.8
TPC-H Query 19	84.5	30.29
TPC-H Query 17	78.57	29.01

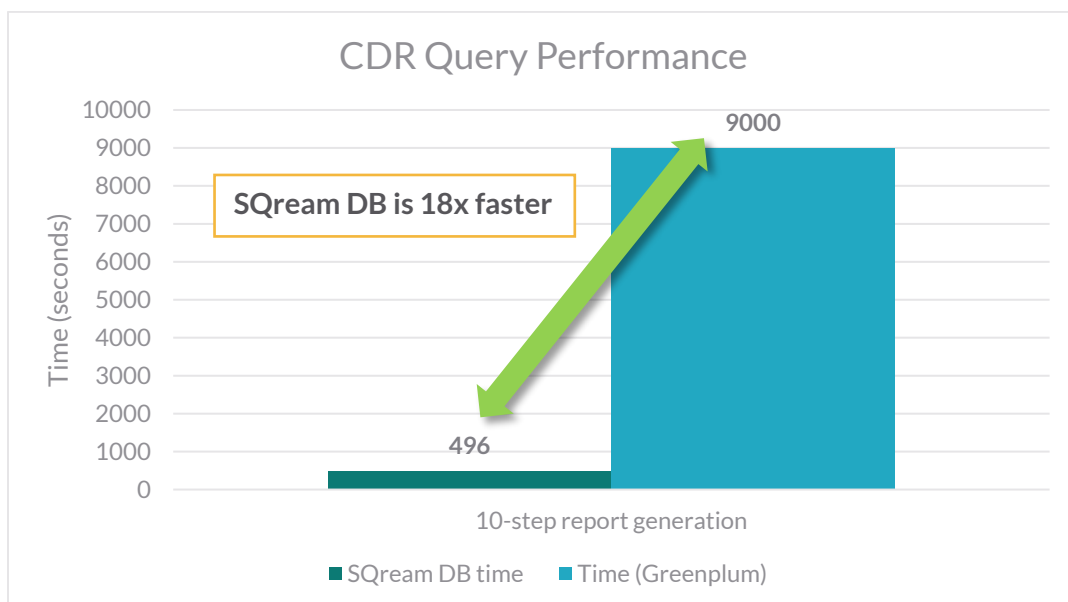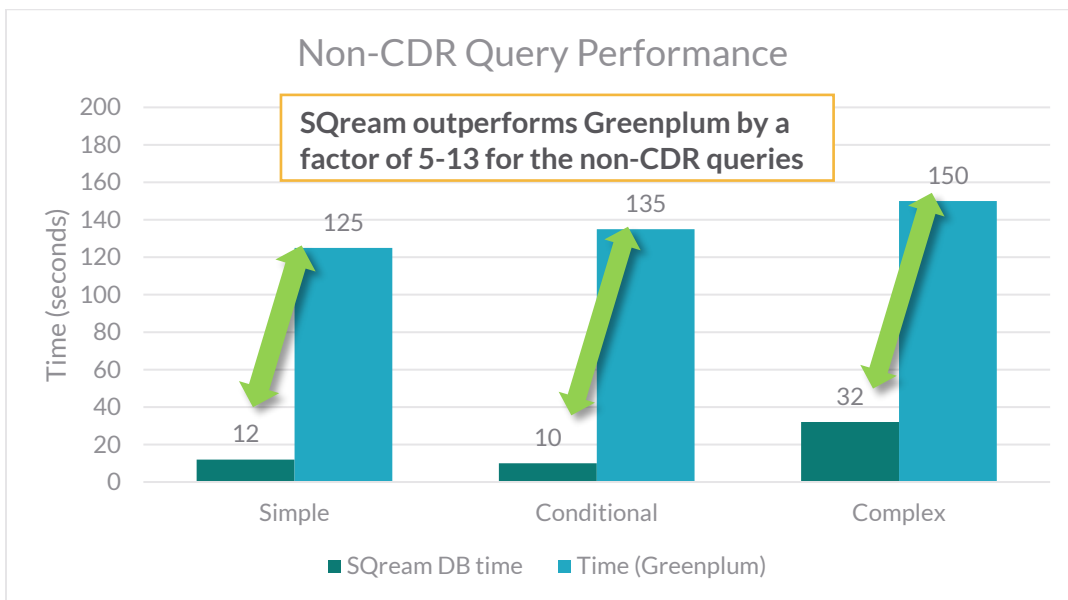■ Industry leading x86-64 server   ■ IBM Power9 AC922

BENCHMARKS

The best way to understand what SQream DB can do for you is to see how fast and cost effective it is in actual customer scenarios, when compared to existing solutions.

Example: 40 nodes of Greenplum at a large mobile network operator

The mobile operator profiled SQream DB in comparison with Greenplum, which consisted of 40 compute nodes in five full racks, with 7600 CPU cores.

The mobile operator wanted to run analytics on a few months' worth of data, at 1.6TB per week. The data represents CDR (call data records) and non-CDR data, such as customer profiles and registered products (i.e. which customer has which phones and subscriptions). SQream DB outperforms the competing system in all situations tested by a factor of 5x-18x, including data ingest, data compression and query performance.

Non-CDR Query Performance

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

	Simple	Conditional	Complex
SQream DB time	12	10	32
Time (Greenplum)	125	135	150

CDR Query Performance

SQream DB is 18x faster

	10-step report generation
SQream DB time	496
Time (Greenplum)	9000

Query types and runtimes

	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 DB 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 DB is a future-proof big data solution.

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

The integration of SQream DB is an easy transition from other SQL databases. There is little-to-no rewriting of SQL queries. SQream DB 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 modelling 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 would normally take to do a few simple variations - and on much less hardware.

SQream DB combines performance, flexibility and ease-of-use, empowering and accelerating your data science and making discovery insights in your data fast, so that you can focus on the core of your business, instead of on the infrastructure.

ABOUT SQREAM

SQream DB combines performance, flexibility, and ease-of-use, empowering and accelerating your data-discovery, so that you can focus on the core of your business, instead of on the infrastructure. Bring the power of SQream DB to your business with a free trial in the cloud or on-premise at sqream.com/try-sqream-db.

Bring the power of SQream DB to your business –

✉ info@sqream.com   🏠 sqream.com   🐦 @SQreamtech