



SQream DB Technical Whitepaper

A database designed for
exponentially growing data

July 2017

THE SHORT VERSION

SQream DB uses GPU technology to improve the performance of columnar queries by at least 20x on large data sets, while reducing the hardware required to perform the query. Typically, a single 2U server equipped with a GPU is equivalent to a 42U rack full of servers.

SQream DB is exceptionally well suited for data science, due to its flexibility. Fast discovery of data science models through reducing query latency allows data scientists to be productive and place models into production quickly.

SQream DB can query large and complex data up to 100x faster than other relational databases.

INTRODUCTION

SQream provides a GPU powered big data analytics solution for data scientists, business intelligence professionals and developers to execute complex SQL queries to derive tactical and strategic insights from 10s of TB to 100s of TB to 1 PB or more of raw data using the tools they use today. SQream’s value proposition is increased productivity, 20x-100x in query performance improvement with a reduced hardware footprint, lower software license fees and simplified administration resulting in more data provided for more insights at better service levels at significantly less cost for complex use cases when compared to currently utilized solutions.

SQream DB enables high velocity querying of large analytical workloads on a single database installation powered by a one or more GPU cards, deployed on-premise or in the cloud.

THE NEED FOR A NEW APPROACH

It is well established that data volumes grow exponentially each year. While data volume grows, the current strategy is to add more nodes, by distributing – either adding more CPU cores, or adding more compute nodes. Because the high volumes of data have out-grown the traditional server capabilities, and modern users and services are highly demanding in both data quantity and velocity – companies tend to compromise on the amount of data and quality-of-services given to these BI professionals and data scientists.

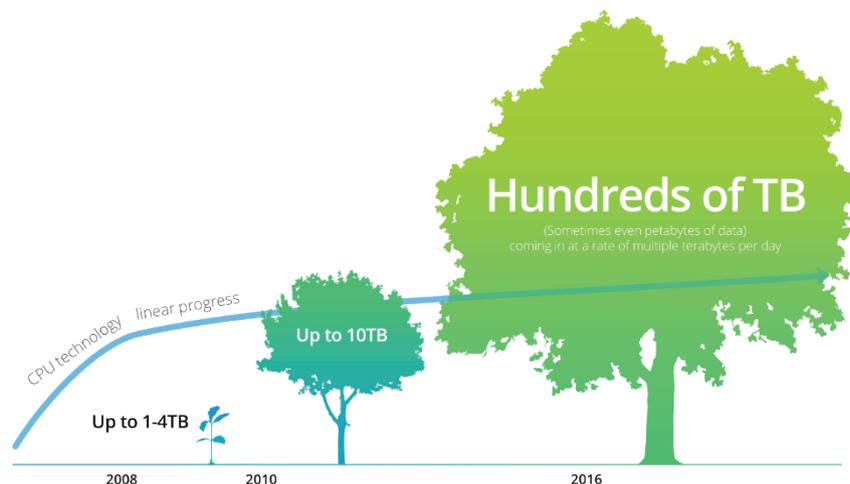


Figure 1 - Advances in CPU technology, contrasted with the growth in data

Typically, an organization might find that they must use several different solutions to address these problems. Some might distribute, resulting in a difficult to manage cluster of servers. With ever-decreasing returns on performance as these clusters grow, the next solution is often a high-performance appliance. However, these tend to have significant costs, with prices in the multi millions of dollars for hardware, software licenses, and skilled professionals.

SQream DB is built-from-scratch to meet these modern requirements – with extreme scalability, easy to use SQL, low-touch tuning and administration, and straight-forward integration with your existing infrastructure.

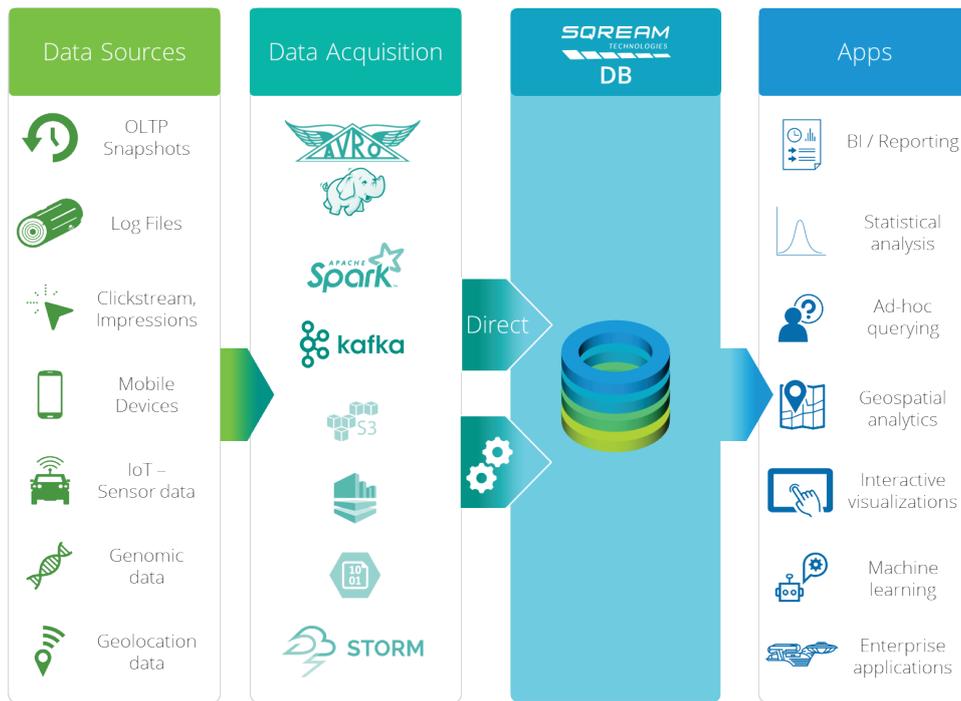


Figure 2 - SQream DB and the variety of data sources and acquisition platforms

QUERY LATENCY AT SCALE

Complex queries contain multiple filters, type conversions, complex *predicates*, exotic *join* semantics, *subqueries* and many of them.

When running this kind of query on large data sets (>100 terabytes in billions of rows in several tables), the number of numerical computations performed is a product of the complexity of the query predicates and the number of rows to be processed.

```
SELECT array_agg(players), player_teams
FROM (
  SELECT DISTINCT t1.t1player AS players, t1.player_teams
  FROM (
    SELECT
      p.playerid AS t1id,
      concat(p.playerid, ':', p.playername, ' ') AS t1player,
      array_agg(pl.teamid ORDER BY pl.teamid) AS player_teams
    FROM player p
    LEFT JOIN plays pl ON p.playerid = pl.playerid
    GROUP BY p.playerid, p.playername
  ) t1
  INNER JOIN (
    SELECT
      p.playerid AS t2id,
      array_agg(pl.teamid ORDER BY pl.teamid) AS player_teams
    FROM player p
    LEFT JOIN plays pl ON p.playerid = pl.playerid
    GROUP BY p.playerid, p.playername
  ) t2 ON t1.player_teams=t2.player_teams AND t1.t1id <> t2.t2id
) innerQuery
GROUP BY player_teams
```

Even when distributed, a conventional query engine using CPUs alone cannot deliver the result within an acceptable period. The query latency is huge, ranging from many minutes to hours.

SQream DB can execute the same query on the same data set with a latency of seconds to minutes.

SQREAM DB – ARCHITECTURE

The idea behind SQream DB’s architecture is harnessing the readily available power of thousands of parallel processing cores in a cost-effective GPU. This power provides a cost-effective alternative to classic MPP (parallel) DBMS solutions, running on hundreds or thousands of expensive general-purpose processors, distributed over dozens of compute nodes.

SQream DB is designed with simplicity in mind, to fit in almost any IT environment, and usable by anyone with even basic SQL knowledge.

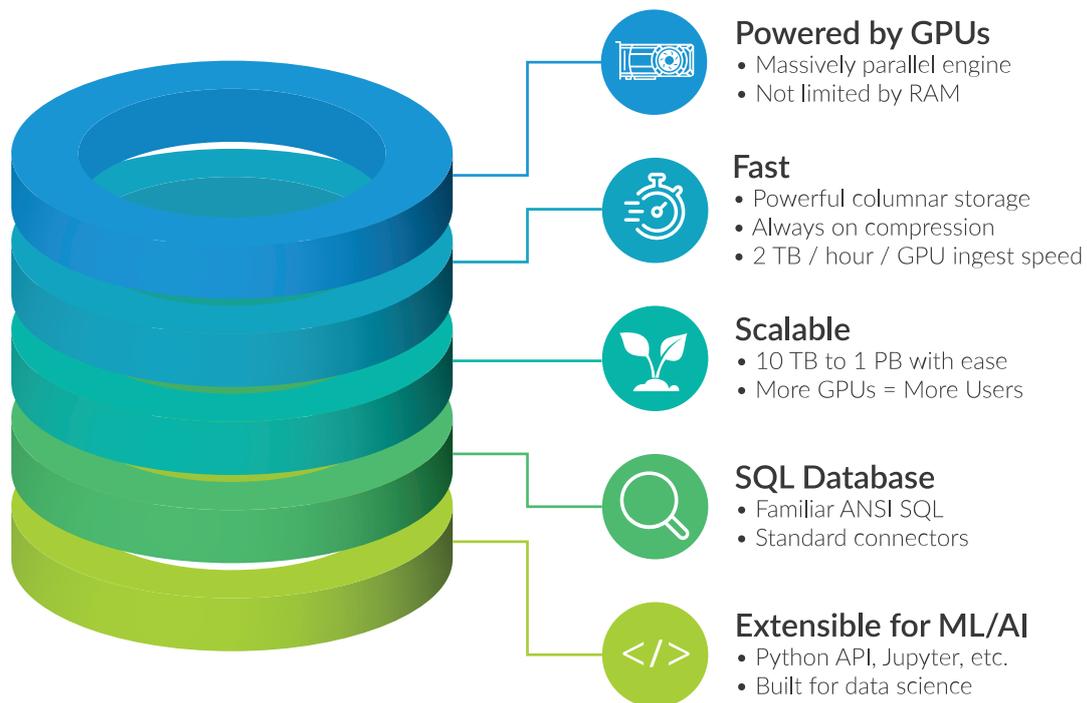


Figure 3 - SQream DB - An SQL database powered by GPUs

DESIGNED FOR AND AROUND THE GPU

A 32-core CPU installation (latency- oriented) requires a lot of power and can cost thousands of dollars. On the other hand, a single throughput-oriented GPU can have as many as 5,000 onboard cores, delivering superior performance at a significantly lower cost, and 90% reduced power consumption.

These GPU processors are specifically designed to perform high-volume and high-velocity numerical computations on both fixed and floating-point values.

Several GPUs can be used inside the same chassis (up to 40), and allows SQream DB to respond to many concurrent client requests.

The high-occupancy nature of GPUs inside a single chassis translates into simpler deployment, reduced networking requirements and smaller overall footprint.

Combining CPU and GPU resources

SQream has patented and proprietary technology that uses a GPU card as a massively parallel processor tailored to process complex SQL queries.

While other, clustered solutions may be massively parallel through scaling-out compute nodes, SQream DB is massively parallel through the GPUs' thousands of on-chip cores.

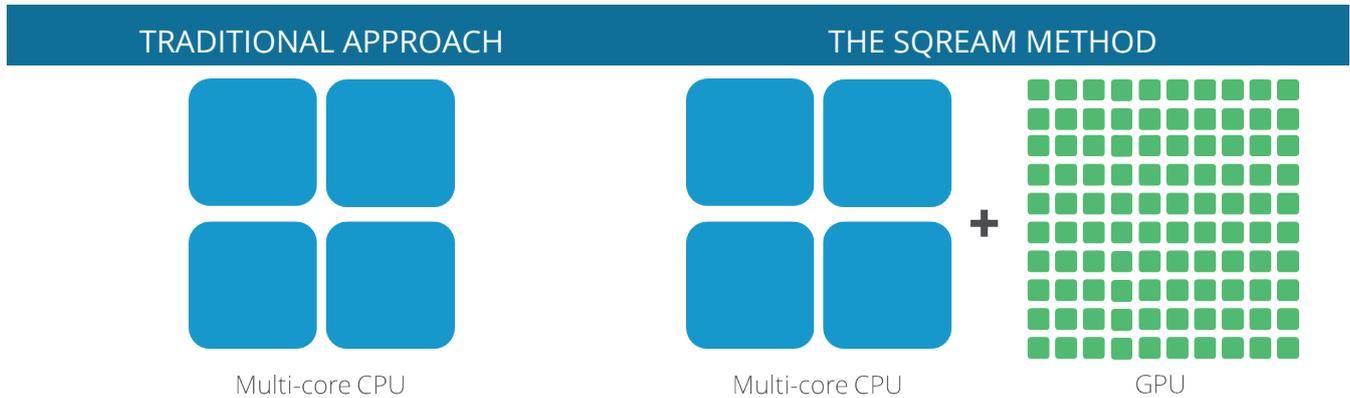


Figure 4 - CPU technology vs. GPU technology

GPUs are not well-suited for handling tasks that cannot be parallelized. They excel at performing (relatively simple), repetitive operations on large amounts of data in many streams. For example, the different architecture of the GPU does make it unsuitable for performing text operations that change based on the content. Because the (thousands of) GPU cores like to work the same way, having contents that will cause the code to behave differently will cause a significant drop in performance – essentially becoming partially sequential and not strictly parallel (in what Nvidia’s CUDA calls branch divergence).

SQream DB makes the most of what resources the host system offers, both CPU and GPU.

For example, the compiler might decide to perform the SQL JOIN operations on the CPU if it feels that the overhead of copying data to and from the GPU would slow down the process.

For text processing, some operations are best performed on the CPU. Therefore, the compiler will decide which columns will go up to the GPU for processing and which will stay on the host and be processed by the CPU.

This balance of CPU and GPU operations is key to ensuring good performance.

POWERFUL COLUMNAR STORAGE ENGINE

COLUMNAR FOR ANALYTICS

Like other analytics databases, SQream DB is a columnar database. This design aspect is well suited for the GPU, because the GPU operates optimally when the data types are consistent. Every column is stored as a collection of “data chunks”, each containing millions of values. SQream DB automates the creation of *smart metadata* on top of each column and every data chunk.

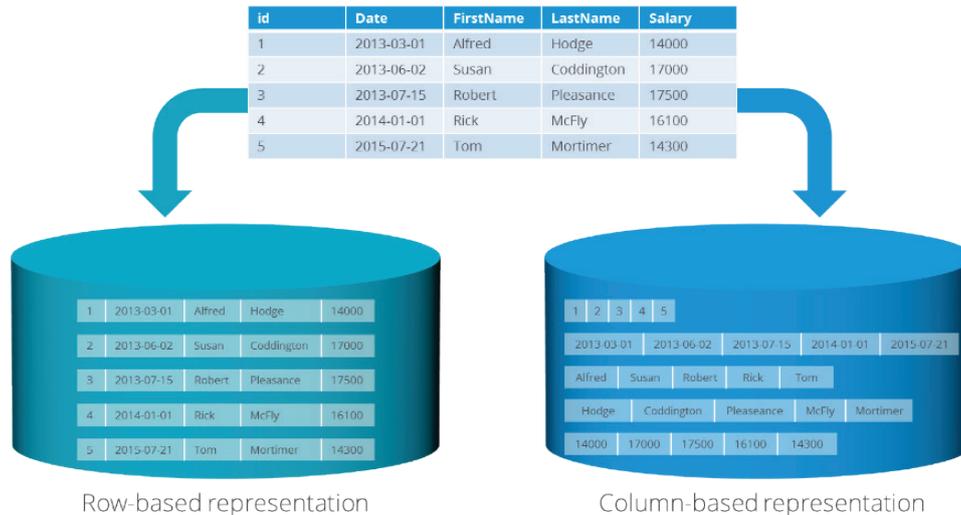


Figure 5 - Row based representation vs. Column based representation

Columnar databases are more efficient when aggregations are calculated over many rows, because scanning only the relevant columns is necessarily faster than reading entire rows of data.

Additionally, columnar databases compress data more easily because of the higher data locality of the columns. Meaning, columns tend to store data that is more similar, and therefore compress better.

AUTOMATIC PARTITIONING

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

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 seemingly straightforward concept enables SQream DB to operate optimally on 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.

SQL COMPILER AND OPTIMIZER

Interacting with SQream DB is no different than interacting with another RDBMS, even though it has many differences under the hood. The SQream DB interface is ANSI-92 compliant SQL (with some extensions added). The query is issued by the user either directly or through a connector like ODBC or JDBC, as well as native Python and ADO.Net.

This SQL command is parsed and converted to Relational Algebra for further processing and optimizations.

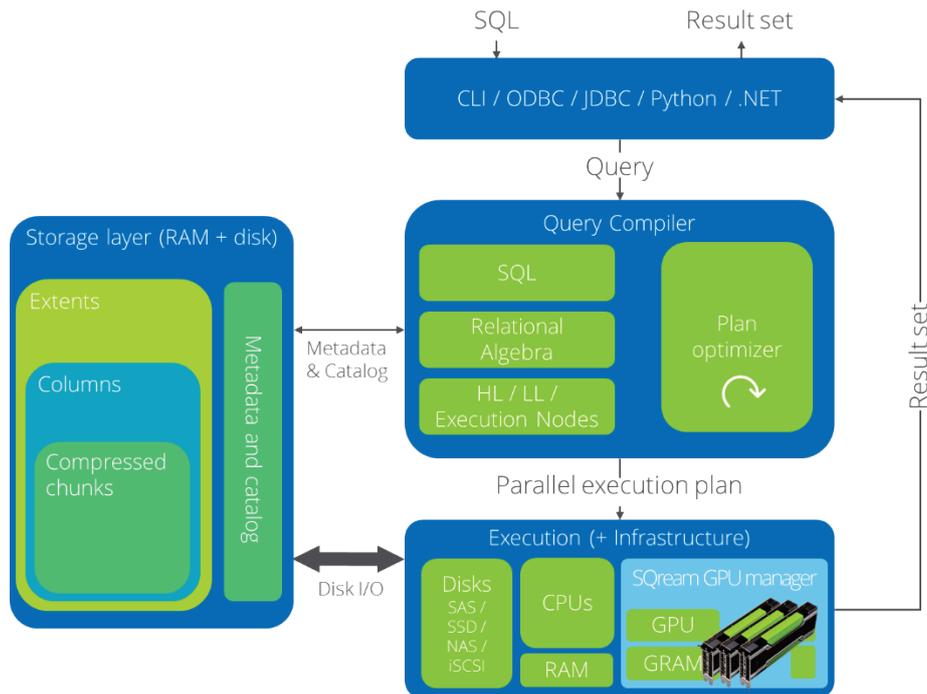


Figure 6 - SQream DB internal architecture

Relational Algebra is a powerful model based on mathematical theory and is used by many SQL engines. The internal operations described as filters and joins, are such strong concepts, that they are comparable to mathematical basics like addition and multiplication. Relational Algebra is therefore not only well studied, but comprehensively battle tested in real world applications. By converting SQL queries into clever, highly parallelizable relational algebra operations, SQream DB can efficiently perform complex operations on the massively parallel GPU cores. These operations are performed internally and automatically by the SQream DB compiler and require no user intervention.

Additionally, our patented and patent-pending optimizations make optimal use of the GPU. For example, SQream DB's algorithms avoid copying unnecessary columns to the GPU, and may swap out some calculations with smaller indexes, in what's known as an 'index gather' operation.

SQREAM DB – FEATURES, INTERFACES AND INTEGRATION

SQL SUPPORT

SQream DB supports an ANSI-92 SQL compliant syntax. SQream DB easily integrates into existing ecosystems by supporting industry standard ODBC and JDBC connectors, as well as Python and ADO.Net. This means existing ETL and applications can stay, minimizing the time needed to get up and running with SQream DB.

You can find more information about the SQL support in the [SQream DB SQL Reference](#).

INTEGRATION

One of the most common tasks for any analytics database is loading data into it from an external source. SQream DB can ingest up to 2 TB per hour per GPU from a variety of sources, either directly from flat-files or through a variety of off-the-shelf tools and simple shell scripts.

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.

IT MONITORING

SQream DB runs on standard hardware, and on standard Linux distributions. 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.

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.

ENTERPRISE FEATURES

OBJECT-LEVEL PERMISSION SYSTEM

SQream DB offers an object-level permission system, with roles and object control all the way down to per-table authentication.

You can find more information about this feature see the [SQream DB SQL Reference](#).

SCALABILITY - OUT AND UP

SQream DB's performance scales linearly as data is added, and additional compute nodes have little to no effect on the performance of queries. SQream DB's algorithms allow it to scale linearly with the size of the data, meaning your queries will not become exponentially longer time as your data grows linearly.

SCALING UP

As easy as adding more drives to your server. Our capable storage architecture takes care of the rest. Because SQream DB is throughput intensive, you can opt for multi-terabyte SAS drives or SSDs, but these are not hard requirements. You may also use any network storage like NAS or SAN, if they can be mounted as a Linux filesystem.

SCALING OUT WITH GPUS

Additional compute power is easy. You will not need to replace the entire server. Plug in additional GPU cards (up to 40 per machine in a high-occupancy chassis), and you are ready to go.

SCALING OUT WITH MORE NODES - INCREASED CONCURRENCY AND HIGH AVAILABILITY

For higher availability of resources, SQream DB can be scaled out. This means multiple SQream DB servers can be connected to the same external storage system, while at any point in time - one or more servers are active. If an active server fails, other servers continue to respond to queries, without any data loss.

SQream DB instances do not need to communicate with one-another, other than against the shared data-store. A load balancer manages queries against the active instances, and will send each query to the next available instance. This setup allows for heavier concurrency by more users. This technology is an inherent part of SQream DB's design.

BENCHMARKS

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

Example – 80 nodes of Greenplum in a large mobile network operator

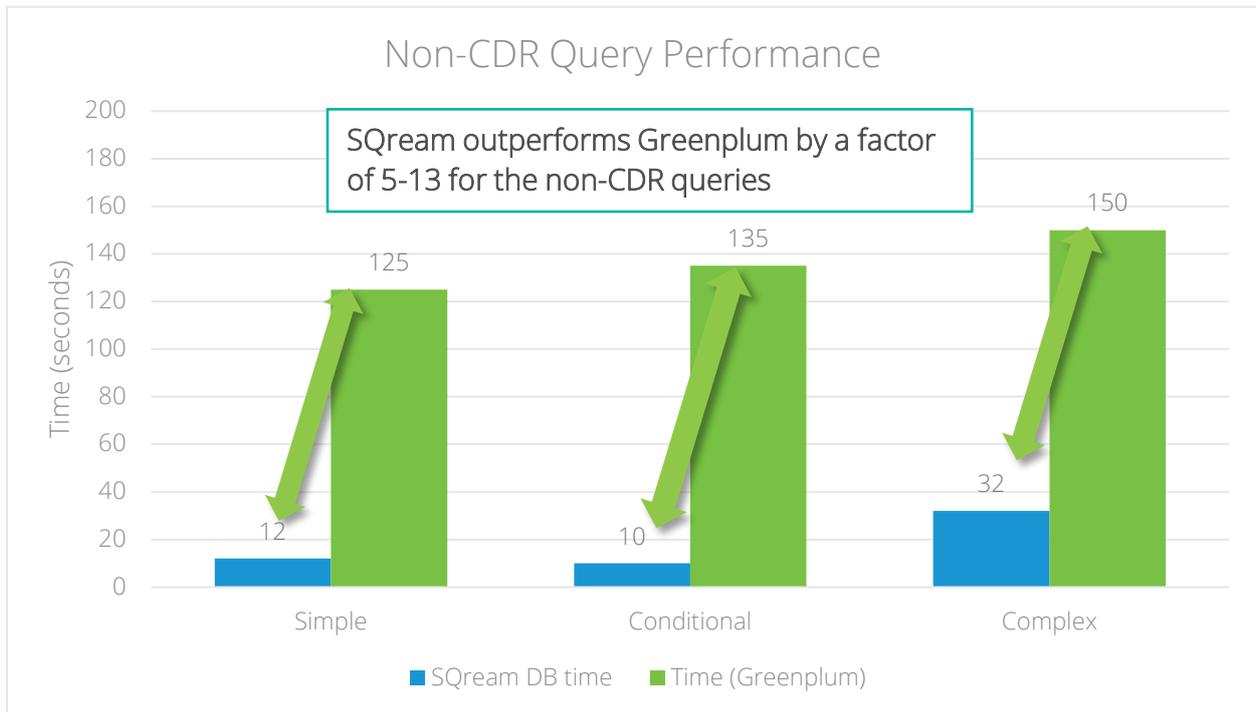
The mobile operator profiled SQream DB in comparison with Greenplum – which consisted of 80 compute nodes in 5 full racks.

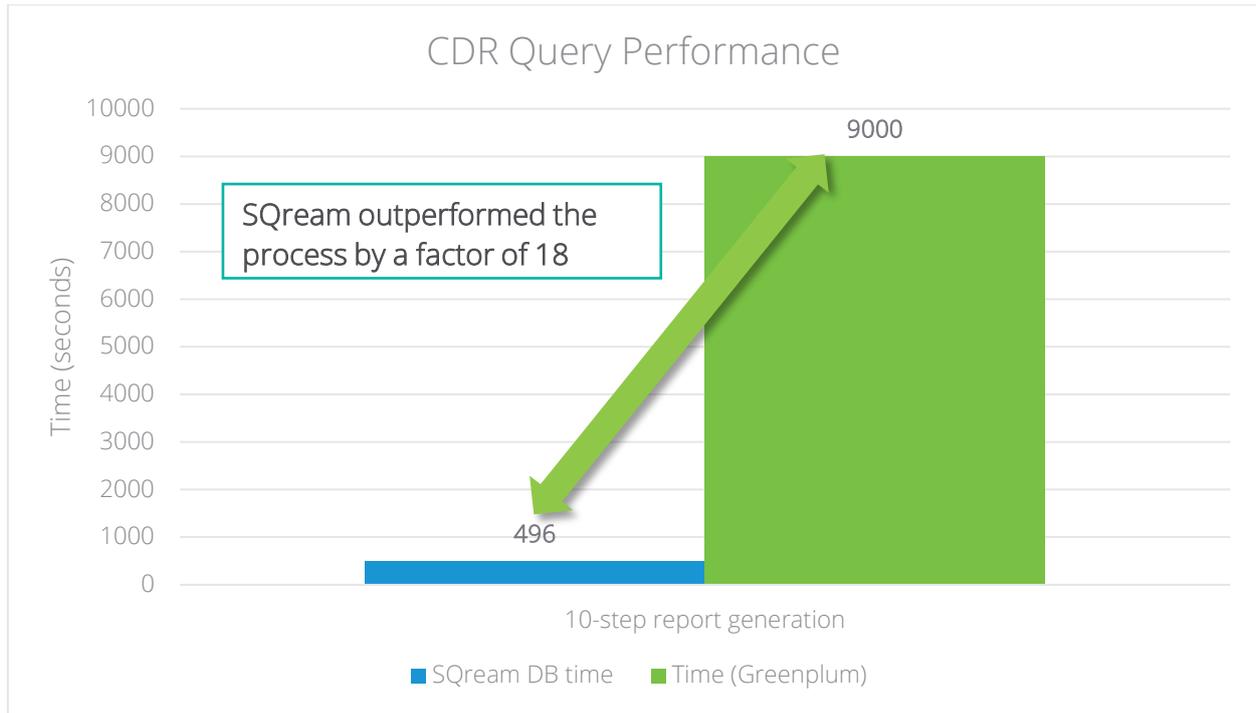
Greenplum is a columnar analytics database, geared towards big data analytics. It is claimed to be one of the most advanced analytical query platform for large datasets.

This 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, like customer profiles and registered products (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.

The server used is a repurposed HP DL380g9, with an added Nvidia Tesla K80 card, further bringing down the project cost.





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 can offer significantly better cost-performance compared to 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 you the opportunity to do much more with more of your data. Getting fast insights with hundreds of billions of data points is now within reach. You may integrate SQream DB 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 plugs in 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 in that they can perform many more variations to the parameters of a model in the same time periods as doing a few simple variations; and on much less hardware.

SQream DB combines performance, flexibility and ease-of-use, empowering your data science and making discovery insights in your data fast, allowing you to focus on the core of your business, not on the infrastructure.

Bring the power of SQream DB to your business –

 info@sqream.com  sqream.com  [@SQreamtech](https://twitter.com/SQreamtech)