GPU Database Acceleration on PowerEdge R940xa

Abstract

This whitepaper looks at the performance and efficiency of GPU database acceleration when using the Dell EMC PowerEdge R940xa server to run Brytlyt GPU DBMS. The objective is to show how the unique CPU to GPU ratio in R940xa is well suited for this new and emerging category of database workloads that leverage the powerful capabilities of GPUs.

August 2018

Revisions

Date	Description
August 2018	Initial release

Acknowledgements

This paper was produced by the following persons:

Author: Bhavesh Patel, Dell EMC Server Advanced Engineering.

Contributors: Richard Heyns, CEO Brytlyt; Palvi Verma, Director of Marketing, Brytlyt.

Table of contents

Executive summary

This whitepaper looks at the performance and efficiency of GPU database acceleration when using Dell EMC PowerEdge R940xa server to run Brytlyt GPU DBMS. The objective is to show how the unique CPU to GPU ratio in R940xa is well suited for this new and emerging category of database workloads that leverage the powerful capabilities of GPUs.

- This whitepaper will discuss some of the background as to why GPUs are becoming the norm in database world.
- How the acceleration of database using GPU can help in some of the emerging workloads like machine learning and deep learning.
- We look at how R940xa architecture with its unique CPU: GPU ratio of 1:1 and support up to 6TB system memory is ideally suited to support GPU DBMS.
- We show how Brytlyt takes advantage of R940xa architecture when running TPC-h benchmark and NY taxi dataset benchmark.
- The paper also shows how you can use R940xa not only to accelerate Brytlyt stack but also run workloads like deep learning by using PyTorch memory management.

1 Evolution of databases

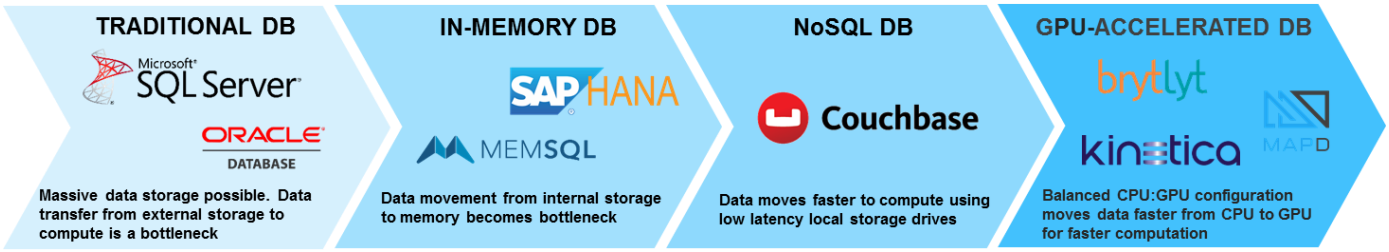


Figure 1 Database Evolution

The business of processing data has been on a continuous evolution and with each advancement there have been newer methods on using different processor architectures in doing database operations.

Earlier on data analysis on servers used storage area networks (SAN) and network-attached storage (NAS) but as the data volume grew, scaling became a bottleneck. This led to using distributed server architecture with DAS (Direct-attached storage) and that is where Hadoop and MapReduce became very popular. This type of architecture is pretty cost effective for batch oriented data analytics but performance is impacted when processing real-time data. In-memory database started to gain traction because servers started supporting RAM (random-access memory) in terabyte range. With higher memory bandwidth and lower latency than DAS, in-memory databases started to gain foothold in database world. The next bottleneck was more in the compute with slowdown in Moore's law and this is where accelerators like GPU started becoming the choice to speed-up data analytics.

2 What is GPU acceleration and how does it apply to databases?

“Think of the GPU as a coin press machine, which can punch out 100 coins with a single operation every four seconds, whereas a CPU is a coin press which can punch out 1 coin per operation every one second. While the CPU has a faster “punch time”, the GPU can punch more coins per minute. This is the key difference between the GPU and CPU. The GPU is throughput oriented, while the CPU is latency oriented.”

The GPU is therefore well suited for operations that perform the same instruction on large amounts of data at once. Put it simply, **a GPU database is a database, relational or non-relational, that uses a GPU (graphical processing unit) to perform some database operations** and because they are throughput orientated they are typically very fast. GPU databases are flexible and can process many different types of data, or much larger amounts of data.

2.1 Why GPU?

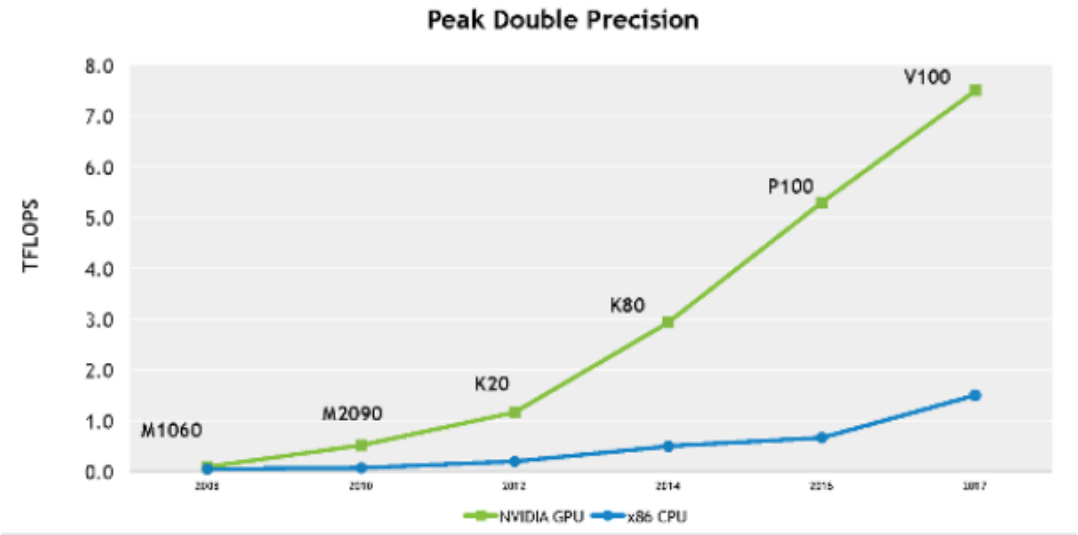


Figure 2 GPU Flop Comparison vs. CPU [source: NVIDIA]

GPUs are highly parallel hardware accelerators originally designed to accelerate the creation of computer graphics. More recently, folks have been looking at GPUs to accelerate other workloads like Database analytics and On Line Analytic Processing (OLAP). Although GPUs are great for accelerating analytics, they have little or no use for transactional (OLTP) style workloads.

GPUs are faster at large numbers of numerical computations than CPUs, whereas CPUs outperform GPUs for tasks that are hard to parallelize or that involve complex control flow instructions. The GPUs small, efficient cores are better suited to performing similar, repeated instructions in parallel, making it suitable for accelerating process-intensive workloads in data analysis applications. Another characteristic of GPUs is their memory bandwidth is much higher, up to 900GB/second when compared to CPUs which is around 68GB/second. The combination of linking together several GPU devices, each with super-fast fast I/O and several-thousand cores means very high rates of single-precision performance can be achieved.

2.2 What database operations can run on GPU?

GPUs achieve their amazing performance by running things in parallel and this means the underlying code must take this parallel way of doing things into account. It also means the algorithms used must be parallelizable, and in many cases parallelizing an operation is not trivial. Relational operations like filtering, sorting, aggregating, grouping and even joining tables are all possible on GPU.

2.3 How do GPUs accelerate analytic workloads?

What makes the GPU in-memory database different to a CPU in-memory database is how it manages storing and processing data to delivering peak performance. Data usually resides in CPU memory in vectorized columns to optimize parallel processing across all available GPUs. The data is moved as needed to GPU memory for both mathematical and spatial calculations, and the results then returned to CPU. For smaller data sets and live streams, the data can reside entirely in the GPU's for even faster processing.

2.4 What are some examples?

a) The U.S. Army's Intelligence & Security Command (INSCOM) unit replaced a cluster of 42 servers with a single server running a database purpose-built to leverage the GPU's power. The application required ingesting over 200 sources of streaming data that together produced some 200 billion records per day.

b) The U.S. Postal Service uses a GPU-accelerated database to track over 200,000 devices that send location coordinates once per minute. In total, the application ingests and analyzes more than a quarter-billion such events in real time every day.

c) A retail company was able to replace a 300-node database cluster with a 30-node GPU-accelerated database cluster. Even with only one-tenth the number of nodes, the new cluster delivers 100-200 times increase in performance for the company's top 10 most complicated queries.

2.5 How can GPU database acceleration help other workloads like Machine Learning and Deep Learning?

Artificial Intelligence is one of the fastest growing segments in the technology space. While only 4% of companies have AI in production right now, over 52% of companies are either starting AI projects or would like to learn more about how AI can be applied to their business. In 2016, the AI market was worth $644 million and the value of that market is expected to grow rapidly, reaching$38.6 billion by 2025. Artificial intelligence will transform the relationship between people and technology, accelerating our creativity and skills.

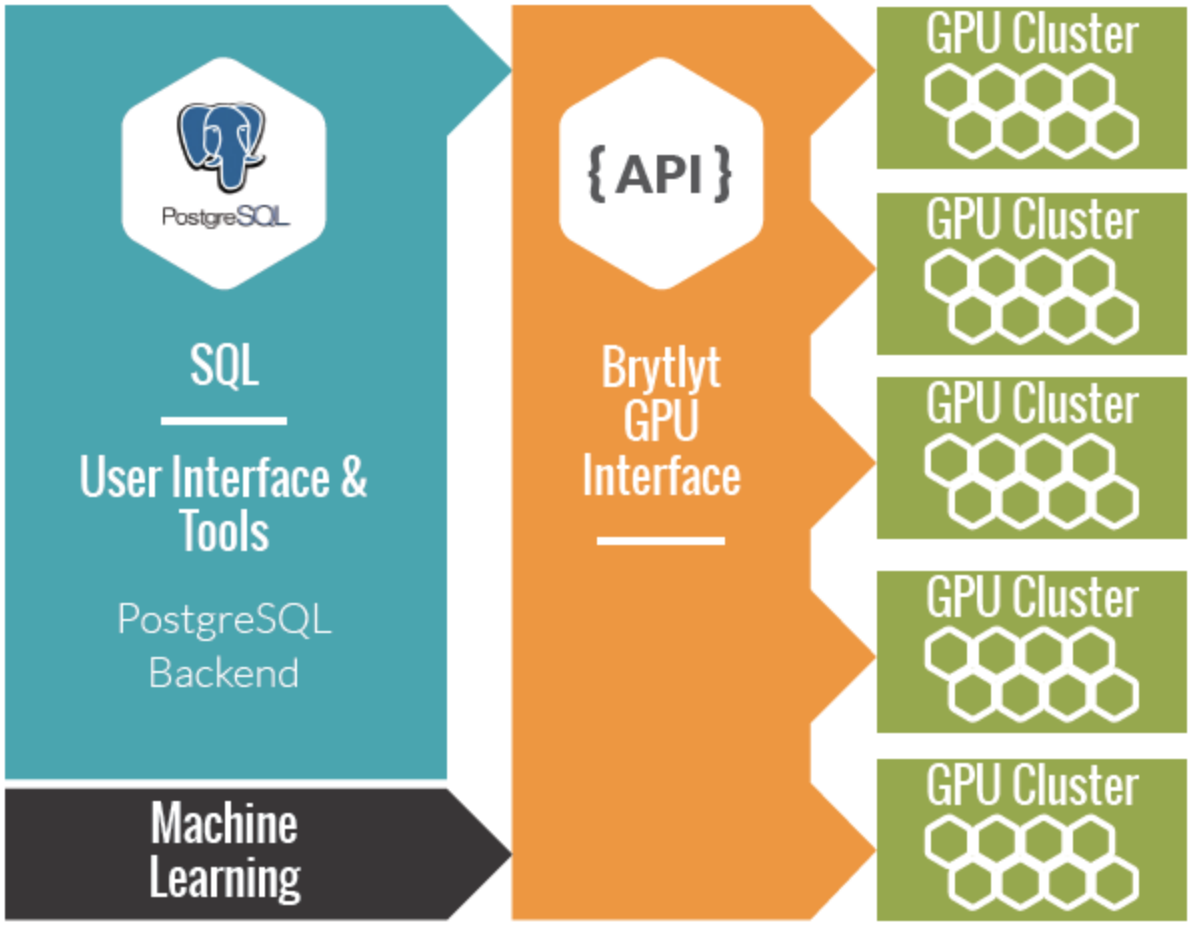


Figure 3 Block Diagram of Brytlyt stack interfacing between PostgreSQL and GPU Cluster

Data Generation involves acquiring, saving, and preparing datasets to train machine learning models. GPU databases offer advantages in all three data generation tasks:

• For data acquisition, connectors for data-in-motion and at-rest with high-speed ingest make it easier to acquire millions of rows of data across disparate systems in seconds.

• For data persistence, the ability to store and manage multi-structured data types in a single GPU database makes all text, images, spatial and time–series data easily accessible to ML/DL applications

• For data preparation, the ability to achieve millisecond response times using popular languages like SQL, C++, Java, and Python makes it easier to explore very large datasets.

2.5.1 How does Brytlyt help with Machine Learning?

BrytMind is an exciting cutting-edge product from Brytlyt which combines **SQL + Artificial Intelligence + GPU** and bridges the gap between tradition SQL, Business Intelligence and data warehousing, and Artificial Intelligence. AI is an umbrella of technologies, from machine learning to natural language processing that allows machines to sense, comprehend, act and learn.

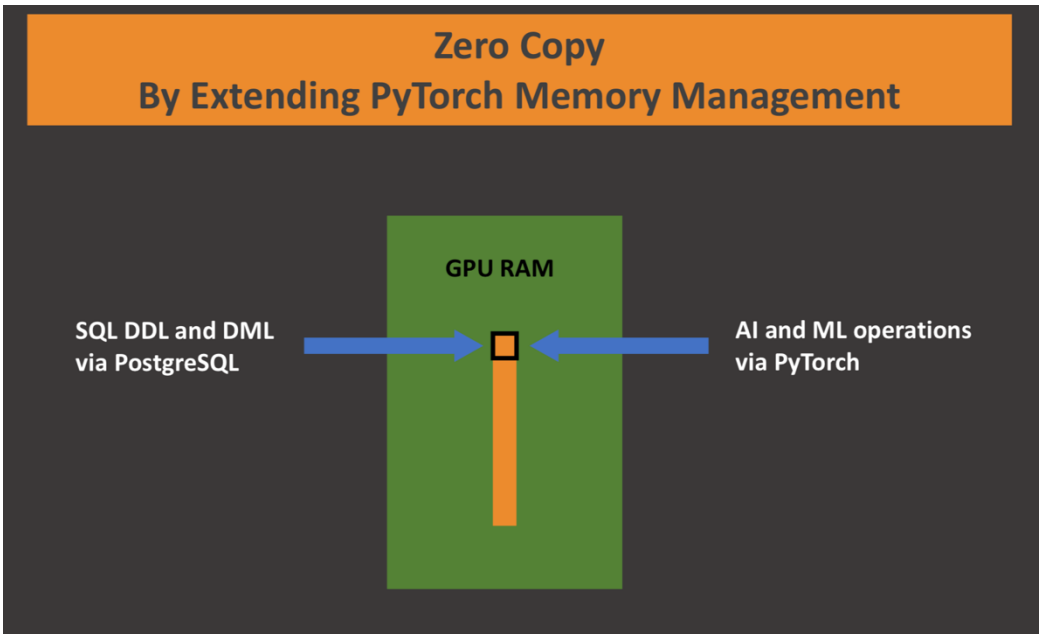
BrytMind uses an enhanced and extended version of PyTorch' s memory management, so there is zero copy required when getting data from BrytlytDB's GPU accelerated database into AI models built using PyTorch running on GPU. GPUs are extremely well suited to Machine Learning and Neural Networks and BrytMind

provides a direct link between the database and the AI models because data already on the GPU is consumed directly by the Artificial Intelligence framework.

PyTorch is an open source machine learning library for Python based on Torch, a scientific computing framework that provides a wide range of algorithms for Deep Learning. It is part of a broader family of machine learning methods that learning using data representations. PyTorch uses a caching memory allocator to speed up memory allocations, and this allows fast memory deallocation without device synchronizations. The net effect is that a column in a BrytlytDB table is exactly the same as a Tensor in PyTorch. BrytlytDB is a fork of PostgreSQL that has been extensively re-written to provide support for GPU acceleration.

Tensors are multi-dimensional arrays that can also be used on a GPU. This means data scientists and analysts can use SQL on GPU for data preparation, and then immediately consume this data directly in Artificial Intelligence and Machine Learning models with zero copy. There is no need to ETL data from the database world to machine learning world.

BrytMind, together with the SpotLyt analytics workbench, brings SQL, visualizations, data workflow, Machine Learning and AI all into on place. And everything is powered by the extraordinary performance of GPUs. With BrytMind, the future of AI promises a new era of disruption and productivity, where human ingenuity is enhanced by speed and precision.



2.6 Why keep CPU: GPU ratio of 1:1 in R940xa?

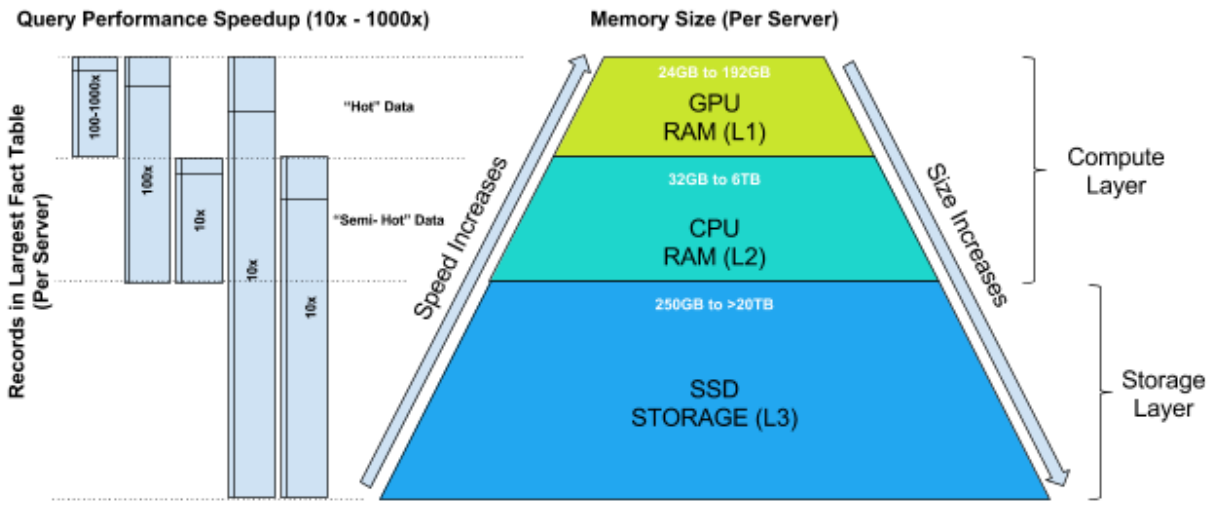


Figure 4 Query Performance with respect to IO bandwidth [Source: MapD]

As shown in Figure 4 above, query performance increases as the data is moved closer to the compute layer. Database acceleration is not achieved if data is fetched from disk (SSD) because of IO bottlenecks. Since GPUs improve performance only when data is available in main memory, any database architecture using GPUs for acceleration should also use CPU in-memory technology.

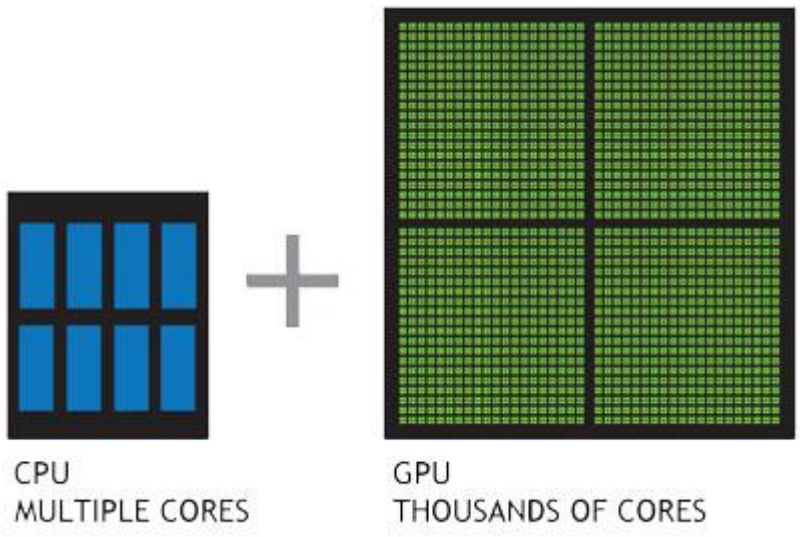


Figure 5 Number of cores in CPU vs. GPU [Source: NVIDIA]

In GDBMS, there are two major IO bottlenecks. The first is the disk IO and second bottleneck is the PCIe bus:

2.6.1 Disk-IO bottleneck

GPUs will not improve performance for disk-based database systems, since most of the time will be spent in disk IO. GPUs improve performance only when data is in main system memory, hence it's much better to keep *hot data* in main memory.

2.6.2 PCIe bottleneck

Data transfers can be significantly accelerated by keeping '*semi-hot data*' in host memory and *hot data* in GPU RAM. But since the GPU RAM is smaller (GBs) vs host memory (TBs), data has to be still transferred over x16 PCIe bus.

Since GPUs have large number of cores, it's faster on certain task like numerical computations than CPUs but GPUs are only faster if the task can be parallelized. In order to avoid PCIe bottlenecks and use the full capabilities of CPU and GPU, it is better to have a ratio of 1:1. This would allow optimal processing for a given operation.

In order to overcome some of the bottlenecks explained in the above sections, we chose a different architecture in R940xa where we wanted to maximize the performance between CPU and GPU & avoid PCIe bottleneck. That is why we kept a ratio of 1:1. We also wanted to have a larger memory capacity, so we could take advantage of both in-memory and GPU, so we could support large databases within RAM and move data into GPU without paying PCIe penalty.

3 The Dell EMC PowerEdge R940xa server

Figure 6 Front and rear views of the PowerEdge R940xa

The rapid increase in machine learning and artificial intelligence applications is changing everything about the way enterprise does business. With a powerful 4-socket and 4U design, the Dell EMC R940xa Server is a great solution to power GPU database acceleration for massive data sets.

The R940xa offers up to 112 processing cores and up to 6TB of memory for consistently fast response times. Add up to 12 NVDIMMs of memory or up to 4 direct-attached NVMe drives to maximize performance and minimize latency.

The R940xa achieves extreme acceleration by combining four CPUs with four GPUs in a 1:1 ratio. Four doublewide GPUs or eight singlewide GPUs or FPGAs can be accommodated.

3.1 CPU: GPU ratio

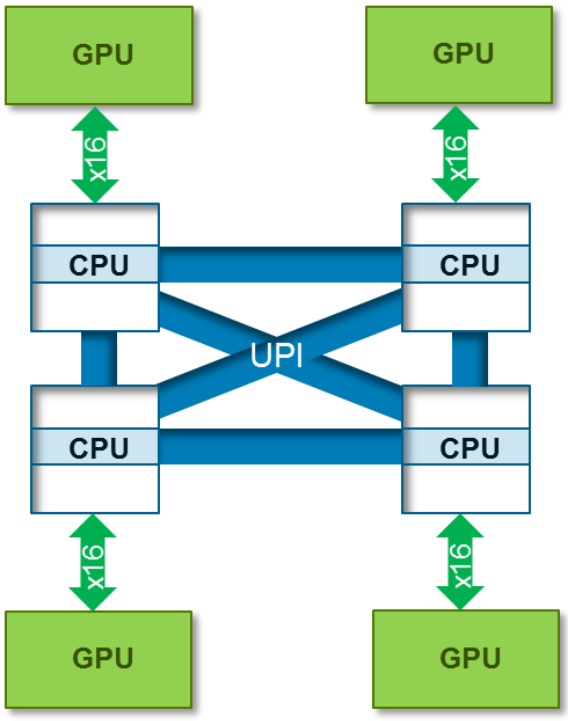


Figure 7 PowerEdge R940xa GPU Topology

4 The challenge with GPU Databases

The biggest hurdle for GPU Databases is to achieve efficient parallel processing for SQL joins. This is crucial, as joining tables is used extensively in industries like Retail, Finance and many more.

The traditional approach for running joins on CPU and is not well suited for the hundreds of thousands of cores in a GPU system. Since GPU's have cores grouped in chunks, with each chunk running the same instructions, most GPU Databases have a tough time with join operations. There is however one GPU Database that has solved this challenge – **Brytlyt.**

4.1 Overview of Brytlyt

Brytlyt is an ultra-high-performance database that combines Graphics Processor Unit (GPU) hardware with its patent pending intellectual property for processing joins in parallel. Brytlyt is built on PostgreSQL and supports all its features, including stored procedures, full JSON support and PostgreSQL's native data connectors. It is easy to leverage existing technology as well as connect directly to virtually any existing data source. This makes Brytlyt easy to integrate with existing investments, lowering time-to-value and increase ROI.

Some highlights:

- **300 times** faster than traditional database technology
- **Real-time analytics** replaces slow batch processing
- **Massively reduce cost** per query by a factor of 10 or more

Brytlyt's Unique Approach to JOINS

Brytlyt has approached the parallelism challenge by devising a patent-pending method that recursively separates rows containing a hit from rows that do not. It breaks the data into blocks and then distributes the blocks to the many cores used for searching.

For example, a dataset of 400,000 rows would be broken into blocks of 200 rows on a 2000-core GPU. Each GPU core then runs its own search on its own block of data in parallel with all the other cores, giving a huge boost in performance over the traditional CPU Database.

Empty blocks are discarded, and the process repeated with the remaining blocks. Then the whole process is done over and over until only the relevant blocks remain. This is an easily scalable process, and the importance of that cannot be overestimated. 10 billion rows could be distributed over 100 GPUs and achieve exactly the same cycle time as 1 billion rows on 10 GPUs.

Brytlyt's solution massively improves data processing power without the corresponding massive financial investment current technology requires.

Comparing Brytlyt performance to other solutions using an industry recognized metric

The TPC-H Composite Query-per-Hour Performance is a metric used to reflect multiple aspects of a database system's ability to process data. Brytlyt performance results were gained from indicative testing based on the TPC-H Query 1. Results shown for other vendors has been sourced from the TPC-H website. In the graphic below, Exasol is recognized as world's fastest in-memory database.

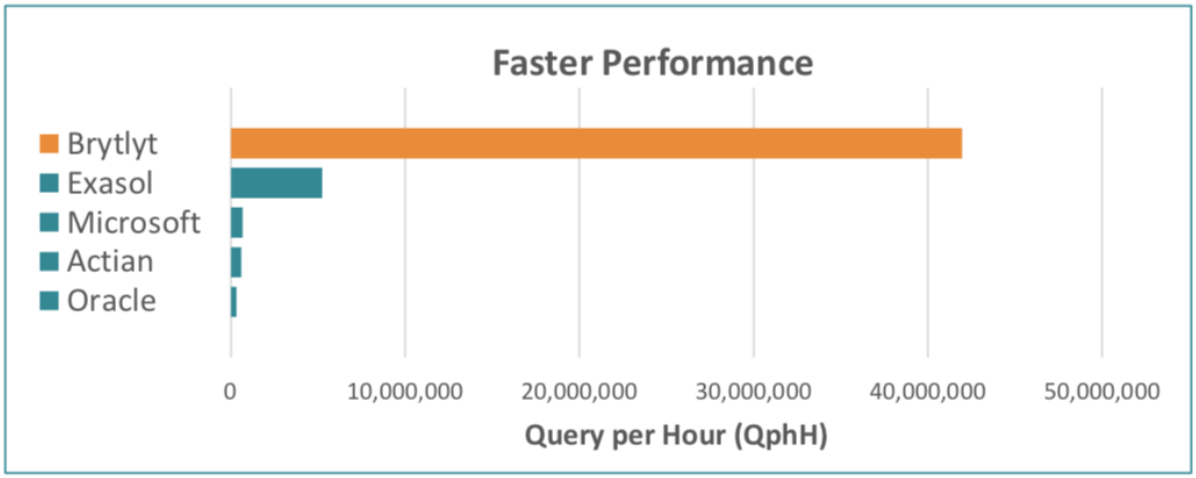


Figure 8 Performance comparison of Brytlyt

4.2 Background on TPC-H Benchmarking [1]

TPC-H is a decision support benchmark and for relational databases with business-oriented ad-hoc queries and data modifications. The data and queries are designed to have broad industry-wide relevance to perform comparative analysis on decision support systems. Large volumes of data are examined to give answers to business critical questions using complex queries and high levels of concurrency. The TPC-H performance metric is called the Composite Query-per-Hour (QphH@Size) and reflects the database size and the query processing power of the system while the TPC-H Price/Performance metric is expressed as $/QphH@Size.

4.3 Importance of TPC-H for Ad-Hoc Business Decision Making Activities

TPC-H is often referred to as the ad-hoc Decision Support (DS) benchmark and is an OLAP workload that measures query analytics in a 'data warehouse' context.

The Decision Support Systems (DSS) are systems that support business and organizational decision-making activities like:

- Business and operational performance indicators and statistics
- Comparison of Sales Figures by time, location and product
- Predicating revenue figures based on sales assumptions
- Evaluating the consequences of different decision alternatives

Decision Support (DS) queries tend to be far more complex, deal with larger volume of data and are therefore far more demanding and long running than transactional workloads like Online Transaction Processing (OLTP).

Because Decision Support (DS) are so complex, it can be extremely challenging for a database designer to plan accordingly and optimize performance and therefore this kind of query may end up running for hours and even days.

The TPC-H benchmark is a well-recognized and highly regarded attempt to model a business database and the ad-hoc Decision Support questions that it must answer. It has been used extensively by both database software and hardware vendors to demonstrate the performance of their solutions, and researchers looking to validate their approaches.

TPC-H is a good test for such systems and most vendors have posted results across a vast range of hardware and data scale factors. TPC-H allows a user to compare database vendors and see how well different solutions perform on different hardware and software. TPC-H can also be used to showcase how well a solution can scale.

5 Brytlyt + Dell EMC PowerEdge R940xa Benchmarking

Extensive benchmarking was done on pre-release Dell R940xa hardware and the results were impressive.

5.1 TPC-H benchmarking on PowerEdge R940xa with Brytlyt

Using TPC-H data and queries, a single Dell 940xa Server with four NVIDIA P100 GPUs was able to achieve through-put of 1.9 billion rows per second at 223 GB/second of raw data for Query 1 and 16.8 billion rows per second at 1.8 TB/second of raw data for Query 6. Join performance was also very good, processing 121 million rows per second when joining the three largest tables (line item 372 million rows, orders 93 million rows, customer 9 million rows).

5.2 NY TAXI Benchmarking on PowerEdge R940xa with Brytlyt

The NY Taxi dataset is made up of 1.1 billion taxi trips conducted in New York City between 2009 and 2015. In CSV format the data is about 500 GB in size.

Benchmarking using the NY Taxi data was just as impressive and there were significant improvements when compared to earlier Brytlyt benchmarking. This was particularly evident for the more complex queries. Because only a single machine with four P100 GPUs was used, the original dataset had to be scaled down from 1.1 billion rows to 243 million rows. Absolute run times were better but as only one machine was used, the overhead of coordinating multiple machines over the network is not included in the runtime.

5.3 PowerEdge R940xa NY Taxi data runtimes

Row Count	243,771,304			
Query	Q1	Q2	Q3	Q4
Run Time (ms)	3	16	99	146

5.4 Previous Benchmark

Row Count	1,100,000,000			
Query	Q1	Q2	Q3	Q4
Run Time (ms)	5	11	103	188

5.5 Reducing Time-to-Value for Data Analysts

This paradigm shift in query performance can massively reduce the time it takes Data Scientists and Analysts to perform queries. For a business this means:

- Giving business leaders better situational awareness on large amounts of high velocity data
- Empowering data scientists to interactively explore large datasets for detailed insights
- Helping data analysts answer crucial business questions in real time.

5.6 Use Cases for Brytlyt's GPU Database

In a world where data driven decision making is more important than ever before, being able to improve time-to-value by answering more complex questions, for more people, on ever growing amounts of data is essential.

Retail Industry

In the retail industry, continuously responding to changing customer needs, in a market where margins are often thin and product cycles are becoming shorter, an accelerated GPU Database like Brytlyt can help:

- Understand customer behavior across social, digital and retail platforms to gain a faster, more accurate 360 view of each customer.
- Understanding seasonal, cultural and regional trends to reveal micro-trends and take advantage of opportunities in real time.
- Understanding inventory at macro and micro level so that products and product lines can respond to customized fulfillment strategies.

Financial Services

Finance is one of the most data driven of any industry, and has a huge amount to gain from Brytlyt's GPU Accelerated Database.

- Risk management calculations on up to the moment data with sub-second speed for better informed investment decisions, react quickly to market events while simultaneously reducing credit risk. By running more complex market or counter party risk calculations, financial industry can obtain results in real time rather than overnight. This speed and quality of information provides a deeper insight to exposures, enabling the financial industry to rapidly adjust positions and reduce risk.
- Fraud detection on large static or streaming datasets to uncover anomalies and patterns that signal fraud while reducing false positives and detecting advanced, persistent threats in real-time.

Telecom Industry

From analyzing network coverage, quality of experience, to making improvements and related investment decisions, and the telecom industry can gain from a GPU accelerate database as Brytlyt:

- Monitoring usage and capacity issues by visualizing real-time data in order to identify the periods of heaviest network usage, telecom providers can use Brytlyt to forecast network capacity, plan for short-term surges in real time and interactively on large and complex data sets.
- Optimizing infrastructure and networks/towers by visualizing the real-time usage for insights on performance, bandwidth or maintenance issues. This can be done in milliseconds and ensuring the health of the networks.

6 References

[1] http://www.tpc.org/tpch/

[2] T. Mostak. An overview of MapD (massively parallel database). White Paper, Massachusetts Institute of Technology, April 2013. http://geops.csail.mit.edu/docs/mapd_overview.pdf

[3] https://spectrum.ieee.org/computing/software/data-monster

[4] https://streamhpc.com/blog/2017-01-24/many-threads-can-run-gpu/

[5] GPU Join processing revisited https://hgpu.org/?p=7692

[6] P. Bakkum and S. Chakradhar. E_cient data management for GPU databases.2012. http://pbbakkum.com/virginian/paper.pdf.

[7] W. Fang, B. He, and Q. Luo. Database compression on graphics processors. PVLDB, 3:670{680, September 2010.

[8] P. Ghodsnia. An in-GPU-memory column-oriented database for processing analytical workloads. In The VLDB PhD Workshop. VLDB Endowment, 2012.

[9] GPU-accelerated Database Systems: Survey and Open Challenges
Sebastian Breß, Max Heimel, Norbert Siegmund, Ladjel Bellatreche, Gunter Saake
https://pdfs.semanticscholar.org/facd/863386054246043a6d0f0aa2baef3d1c806d.pdf

[10] Accelerating SQL Database operations on a GPU with CUDA
https://www.cs.virginia.edu/~skadron/Papers/bakkum_sqlite_gpgpu10.pdf

[11] Database and Hardware: The Beginning and Sequel of a beautiful friendship.
http://www.vldb.org/pvldb/vol8/p2058-anastasia.pdf

[12] Mixing multi-core CPUs and GPU for Scientific simulation software. K.A Hawick
https://pdfs.semanticscholar.org/7415/6444f07f765c47cf5202d85b9156db6041a6.pdf

[13] HippogriffDB: Balancing I/O and GPU Bandwidth in Big Data Analytics_
Jing Li HungWei Tsengy Chunbin Lin Yannis Papakonstantinou Steven Swanson
Department of Computer Science and Engineering, University of California, San Diego
http://www.vldb.org/pvldb/vol9/p1647-li.pdf

[14] Hardware Acceleration of Database Analytics
Evangelia Sitaridi Amazon Web Services

[15] Hardware Acceleration of Database Operations
Jared Casper and Kunle Olukotun
https://ppl.stanford.edu/papers/fpga14-casper.pdf