# PUSHING DATABASE SCALABILITY UP AND OUT WITH GPUS

September 22, 2016 Timothy Prickett Morgan

What is good for the simulation and the machine learning is, as it turns out, also good for the database. The performance and thermal limits of traditional CPUs have made GPUs the go-to accelerator for these workloads at extreme scale, and now databases, which are thread monsters in their own right, are also turning to GPUs to get a performance and scale boost.

Commercializing GPU databases takes time, and Kinetica, formerly known as GPUdb, is making a bit of a splash ahead of the Strata+Hadoop World conference next week as it brags about the performance and scale of the parallel database management system that it initially created for the US Army and has commercialized with the US Postal Service.

Kinetica joins MapD, which we profiled recently, and Sqream Technologies, which you can find out more about here, in using GPUs to execute the parallel functions of SQL queries to massively speed up the processing of queries against databases. Each of these GPU databases has come into being through a unique path, just like the pillars of the relational database world – Oracle's eponymous database, IBM DB2, Microsoft SQL Server (and its Sybase forbear), MySQL, and PostgreSQL – did decades ago. And as these GPU databases mature and develop, the race will be on to scale them up and out to handle ever larger datasets and perform queries faster and faster at the same time.

The GPUdb database was developed by Amit Vij, CEO of GIS Federal, and his long-time partner Nima Negahban, chief technology officer at the geospatial information systems and data analytics firm that, as the name suggests, does a lot of work with the US Federal government. Negahban was the original developer of the GPUdb database that is now being commercialized by a separate company, called Kinetica, which is also the name of the database now.

The database that is now called Kinetica was created from scratch, Negahban tells *The Next Platform*, with its own SQL engine and distributed processing framework and it does not rely on any open source projects. GIS Federal was commissioned by the US Army Intelligence and Security Command at Fort Belvoir outside of Washington, DC, to create the GPUdb database in 2010 after coming to the conclusion that the existing relational, NoSQL, and NewSQL options were not going to be able to handle the data flows and query complexity and volume that searching for terrorist activity required.

"The Army was trying to create a data lake, although that term was not really used at the time, and consume over 200 different data feeds," explains Negahban. "They wanted to bring this data into a common compute capability and give one single API to a community of analysts and developers so they could work together and iterate queries in an agile fashion against real-time data."

GIS Federal looked at the HBase database layer for Hadoop (created by Facebook), the Cassandra alternative to the Hadoop Distributed File System (also created by Facebook), and the MongoDB NoSQL database as possible platforms on which to build this data lake for the Army. While testing these platforms out, Negahban says the same patterns happened again and again: The query inventories were extremely limited, and developers had to go back and create more and more indexes for these databases.

"The cycle kept repeating itself, and in the end, whether it was HBase, Cassandra, or MongoDB, the hardware fan-out grew exponentially as query demand grew, and to have more query flexibility they needed more elaborate indexes, and as every record came in, it had to be indexed in a multitude of ways and therefore the staleness off the queries grew and grew. Initially, as the new data came in, the results were an hour late. And as more indexes had to be built, the delay went to 24 hours. And finally it grew to a week. There was no way the Army was going to get to its real-time goal."

So Negahban and Vij cooked up GPUdb, which was at the heart of the GAIA system created for the Army's INSCOM agency to take in massive amounts of data to help the Army move troops around theatres of operation as safely as possible as threats come and go. The initial Army system running the first GPUdb implementation, which was installed after two years of development in 2012, was a four-rack UV2000 shared memory system with 2,048 processor cores and a 10 TB in-memory database that was accelerated by sixteen Nvidia Tesla K20X GPUs; this system ran Hadoop and HBase was used as the database layer, with offloading of SQL functions to GPUdb.



The smart city of FaaS

Accelerating Solutions for the Datacenter

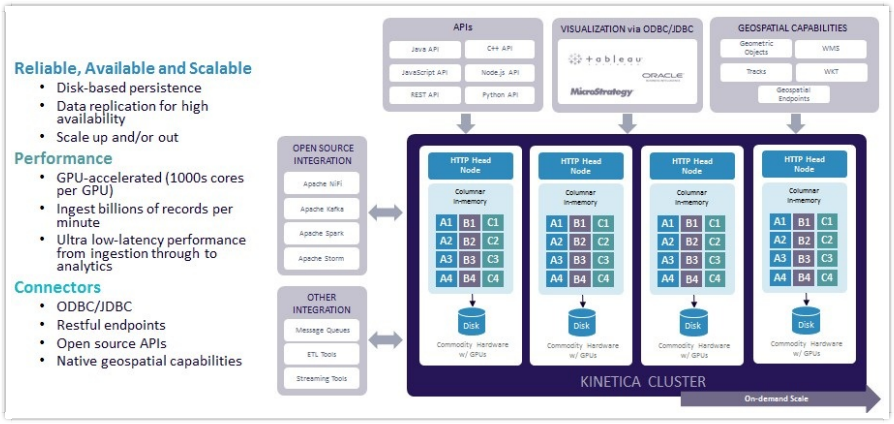
Rethinking System Architectures to Improve Performance and Power Efficiency

## THE NEXT PLATFORM WEEKLY

Tap the stack to painlessly subscribe for a weekly email edition of The Next Platform, featuring highlights, analysis, and stories from the week directly from us to your inbox with nothing in between.

Two years later, GPUdb had its initial commercial installation when the US Postal Service used a more modern version of the database to track its fleet of vehicles and its employee base in real time, and by then, the database was increasingly resident on the GPUs themselves. The Postal Service used one of the famous enterprise databases (it can't say which one) , and wanted to put all of this streaming geospatial data into that relational database, but it was crazy expensive, according to Negahban. Last year, the USPS delivered 150 billion pieces of mail, and the routing software that used the GPUdb as its data store tracked every truck and every piece of mail, mashed it up with environmental and seasonal data, and helped the organization drive 70 million fewer miles than it did in 2015 and save 7 million gallons of fuel.

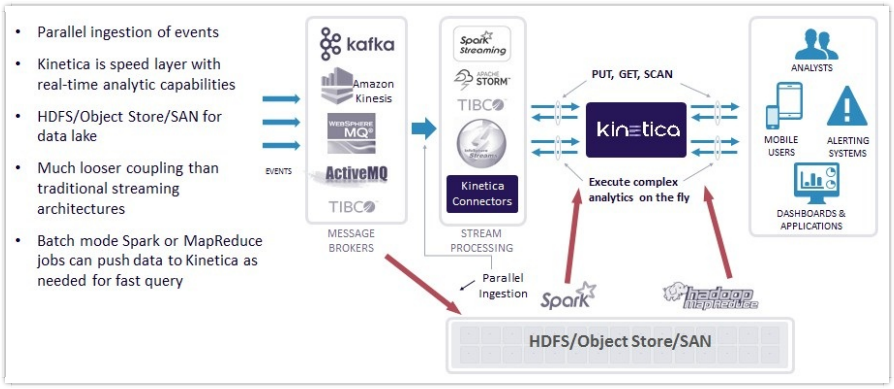
The amazing thing is that all of this was done on a fifteen-node hybrid CPU-GPU cluster. (Well, to be more precise, it was done on a cluster with five-way replication and a total of 72 nodes.) Over 200,000 devices are streaming data into this cluster once every minute, and over 15,000 sessions running queries off this system can be handled simultaneously.




GPUdb and now Kinetica are not open source, and it is not clear they will ever be. The new name suggests that the company is positioning itself to run on any massively parallel device, not just GPUs, but this is not a new idea, but a return to its original plan.

Negahban says that Nvidia's GPUs – in their various incarnations and not always Tesla compute engines, either – are the preferred motors for its database. Any Nvidia GPU that has a Fermi or later generation of GPU (basically from 2009 forward) and that supports the CUDA programming environment can run the GPUdb/Kinetica database.

Technically, any device that supports OpenCL could also run it. (The early editions of the GPUdb software could be deployed on AMD GPUs as well.) But Negahban points out that in the past couple of years Kinetica has done a lot of work optimizing for Nvidia GPUs, and adds that for many customers, double precision floating point math matters for their analytics – actually, it is the format of data that they have that drives this – so the appropriate Tesla accelerators like the K40, the K80, and now the P100 are usually recommended.

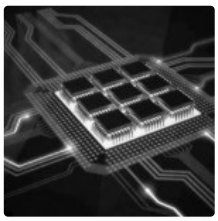



"With the Power systems from IBM and NVLink connecting them to the Tesla P100s, we are going to be able to have much greater throughput to the GPU, and beyond the higher memory bandwidth of the Tesla P100, the Pascal GPUs are just faster processors than other Teslas, too," says Negahban. "We have a unique capability with the Power platform for sure, but we are excited about the P100 on both Power and Xeon platforms."

As far as we know, MapD and Sqream do not yet support Power processors, but there is no reason they could not. GIS Federal is a member of the OpenPower Foundation and was a beta tester for the Pascal Teslas, so Kinetica has been looking forward to this day when NVLink was available. (IBM just started shipping systems that support NVLink.) Kinetica has also been working on integrating flash devices with the hybrid Power8-Tesla compute complex through the Power8 chip's Coherent Accelerator Processor Interface (CAPI) ports, and we could see a day when disks are replaced with flash and the whole shebang moves that much faster.

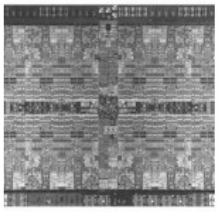
Kinetica hosts the data that drives analytics in both disks in a traditional row format and in CPU memory in a compressed columnar format like other in-memory databases. The secret sauce in all of this is the memory management as data moves from disk to memory and back. The authoritative

## SIMILAR VEIN

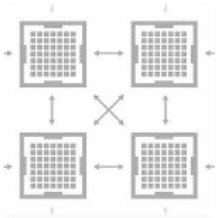


OpenPower: Accelerated Computing Will Be The New Normal

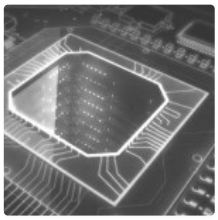
Refreshed IBM Power Linux Systems Add NVLink



IBM Unfolds Power Chip Roadmap Out Past 2020



NVLink Takes GPU Acceleration To The Next Level



Future Systems: Pitting Fewer Fat Nodes Against Many Skinny Ones

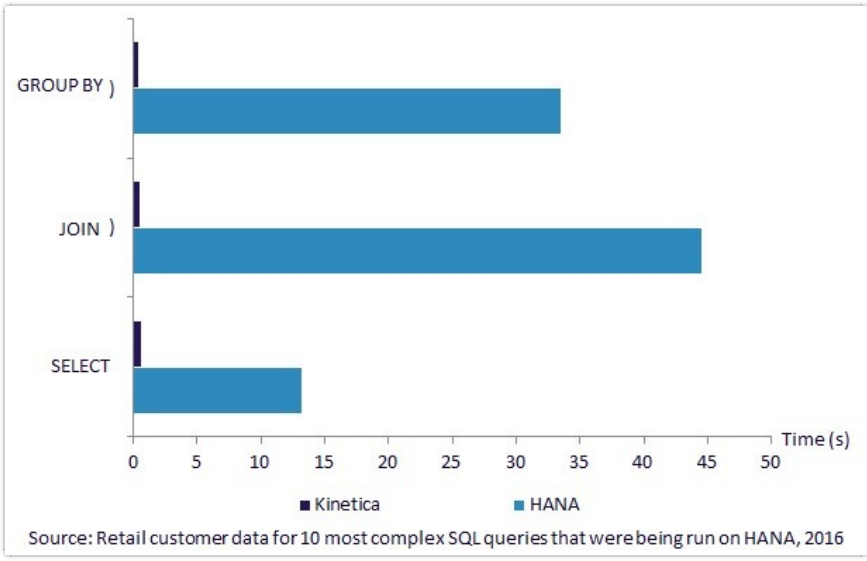
GPU Accelerators Radically Boost SQL Queries

copy of the database lives in the server host memory, and interestingly, the GDDR or HBM memory of the GPU is used as a scratch pad for data as the GPU chews it. The datasets are so big that Kinetica is dealing with that cramming it into the 4 GB to 24 GB of GPU memory was not going to work; they demand terabytes of memory per node, not gigabytes, and ganging up multiple GPUs is too costly. The memory to compute ratio is not good, so in effect, the CPU RAM is a memory offload to the GPU processing as far as Kinetica is concerned. (Funny, that.) The point is, these customers need billions to hundreds of billions to even trillions of rows of data, and that is not going to fit in GDDR or HBM memory, and even if it did, it would be too costly to scale out the GPUs.

Kinetica has about a dozen customers in production, with the largest one (unnamed) having a cluster with 60 nodes. The architecture is designed to scale out to hundreds of nodes, and with the combination of Power8 and Pascal Teslas with CAPI flash storage, an extremely brawny node could be made and then clustered. (Think of the "Minsky" Power Systems LC machine IBM just launched or Nvidia's DGX-1 machine running this GPU-accelerated database.)

Not that you need big iron to run the Kinetica database. A two-node Supermicro cluster, with each server having two 12-core Xeon E5-2690 processors running at 2.6 GHz plus two Tesla K80s, 512 GB of main memory, and a 3 TB SSD in each node can query a database of 15 billion Tweets and render a visualization of that data in under 1 second. That will run you about $50,000, not including the Kinetica software.

Here is how a 30 node cluster of Xeon CPUs and tesla K80 GPUs running Kinetica 5.2 against a data warehouse made up of more than 100 nodes of SAP's HANA in memory database:



Source: Retail customer data for 10 most complex SQL queries that were being run on HANA, 2016

You can't even see the lag with the Kinetica database. It is well under sub-second response time. And without being specific about pricing – which is done on the basis of the capacity of the database, not the compute power behind it – Kinetica says that the performance is orders of magnitude better than HANA or MemSQL and orders of magnitude less costly at the system level.

One important consideration for enterprises that are knee deep in SQL queries is that the Kinetica 6.0 database release, which will be coming out later this year, will have full compliance with the ANSI SQL-92 standard, which means it speaks the same SQL that traditional relational databases have for a long time. The prior GPUdb 5.2 release, which came out in May and which was used by the USPS in its cluster, was about 70 percent compliant with the SQL-92 standard, which is about as good as some Hadoop overlays do but it is not the same as an enterprise-grade relational database management system with full compatibility.

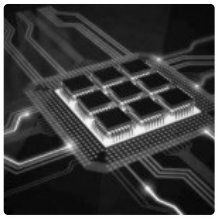
"Simple SQL queries are easy to do, and even moderate ones are not that hard to be compliant with," says Negahban. "But we want to get to the point where you can take a large, complex 500 line query and paste it in and run it without any problems. Even though we have native APIs with a lot of languages, at the end of the day, SQL is still is the gold standard in OLAP processing and reporting. So it is mission critical for us to get that full compliance."

That is precisely what $7 million in first round venture funding, which Kinetica scored in April, is for.

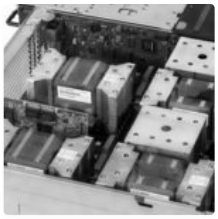
SHARE THIS:

Reddit Facebook 24 LinkedIn 45 Twitter Google Email

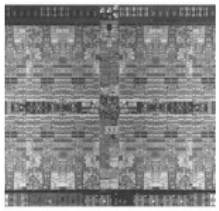
## SIMILAR VEIN



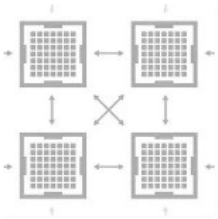
OpenPower: Accelerated Computing Will Be The New Normal



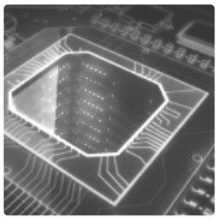
Refreshed IBM Power Linux Systems Add NVLink



IBM Unfolds Power Chip Roadmap Out Past 2020



NVLink Takes GPU Acceleration To The Next Level



Future Systems: Pitting Fewer Fat Nodes Against Many Skinny Ones

GPU Accelerators Radically Boost SQL Queries

Categories: Analyze, Compute, Enterprise

Tags: GPUdb, IBM, Kinetica, Nvidia, NVLink, Power8, Tesla

IBM Builds A Bridge Between Private And Public Power Clouds

## 4 THOUGHTS ON "Pushing Database Scalability Up And Out With GPUs"

*Jozo* says: September 23, 2016 at 2:28 am

Nice to see, that somebody finaly put some serious effort into this area.

Btw.: how it is compared to DAX engines implemented in SPARC M7?

Reply

*OranjeeGeneral* says: September 23, 2016 at 6:37 am

Interesting but I wonder if these GPUdb wouldn't actually benefit more from a XeonPHI architecture, with its vastly better integer performance, FLOPS don't give you much in DBs. Better and faster memory and closer to the main memory as it can act as the host CPU as well.

Reply

*Jozo* says: September 23, 2016 at 9:53 am

KNL XeonPHI is oriented at flops too. It has 2 very wide vector procesing units capable of 64Flops per cycle (via FMA instructions) but only 2 scalar ALUs, which yields only at around 256 "Giga integer ops per second" (72 cores at 1,5 GHz), which is actually much lower than what people measured on gaming-class GPUs.

And this number will be even lower because of low ILP database code. XeonPHI has too small reservation stations so it can't exploit enough paralelization in code.

So if you want Intel with best integer performance, then you should use regular Xeons. 😉

Reply

*jimmy* says: September 23, 2016 at 5:37 pm

What vastly better integer performance?

GP100 has 1:1 FLOP:INTOP performance…

Basically if GPUs are already whopping Xeons, dont bring a crappy Xeon Phi to the table and expect it to do any better.

Reply

## LEAVE A REPLY

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

Post Comment

## PAGES

About
Contact
Contributors
Newsletter

## RECENT POSTS

Pushing Database Scalability Up And Out With GPUs
IBM Builds A Bridge Between Private And Public Power Clouds
Baking Specialization into Hardware Cools CPU Concerns
The Three Great Lies of Cloud Computing
Modern Storage Software Erodes Resistant Data Silos

Copyright © 2015 The Next Platform