Intelligent and Hyper-Fast Access to Data

Technology of Levyx’s Core Data Engines Helium™ and Xenon™

January 2017

In the past, real-time processing of data has meant deploying cumbersome (i.e. inefficient and complex) and expensive memory-based deployments of systems and in-memory databases which were needed to keep up in these rigorous environments. Not anymore.

Levyx’s software fundamentally disrupts the economics of Big Data processing in real-time, bringing the benefits of affordable, high-speed data processing to an expanded variety use cases and applications. Levyx’s Helium™ solution offers a high-performance key value storage engine that enables input/output (I/O)-intensive legacy and Big Data applications to operate faster, simpler, and cheaper. Xenon™ extends the technology further making it applicable to distributed environments thereby expanding its usefulness to broader set of applications, including the most prevalent and user-friendly Big Data platforms on the planet.

Our Primary Innovation and Why It Matters

Currently accessing, processing and ultimately deriving value from large scale data sets is limited as the current techniques for this are not scalable, are highly inefficient in utilizing resources, or are quite simply not fast enough. Based on this premise, Levyx has developed a simpler, more scalable I/O stack - one that fundamentally rethinks of the “Traditional” data path and creates a new one that is more relevant to (i.e. designed for) today’s most advanced datacenter architectures and the hardware innovations that underlie them in order to fully maximize them.

In the traditional data path, data must travel through main memory, the I/O subsystem and into and out of the storage media (e.g. Flash). Flash and non-volatile memories (NVMs) are not optimized and files systems and OS kernels do not fully utilize the available bandwidth. As a result, block-oriented unstructured access is highly inefficient.

On the other hand, the Levyx data path is characterized by a single, persistent, high-capacity, high bandwidth, low-latency memory layer that is scalable with the number of cores in a system,
cluster or network, and with the bandwidth of the IO system (SSD or Storage Class Memory). Under our schema, the benefits of Flash/NVM are fully exploited, either in single node or distributed system, and the result is a single-persistent memory layer that is object-oriented and allows for highly structured data access.

We achieve this new design by integrating high-capacity Flash and next-gen NVM fabrics into the system memory hierarchy. In addition, we employ rich and expressive KV semantics instead of block offering a native match to Big Data, HPC, and IoT application data demands. Our proprietary indexing methodologies are massively parallel, compressed, memory-resident, lock-free indexer, and can track billions of ordered objects with sub microsecond lookup latencies. Since we do object-level caching of data throughout the memory hierarchy, we can access petabytes of persistent storage at main-memory speeds and latencies. Finally we apply seamless and efficient translation of application-level parallelism to the physical HW processor, I/O bus, and Flash/NVM channel bandwidths.
System builders, drive manufacturers, and chip vendors (hardware guys) can derive benefits from this technology as a critical building block for developing highly-dense platforms loaded with Flash that provide their customers with more cost-effective alternatives to DRAM-laden systems whose scaling inefficiencies make them unwieldy. Flash arrays that fail to deliver on the promises of performance advantages, or expensive proprietary platforms that are simply not affordable.

For end-users the opportunities for Levyx are wide-ranging as the technology is relevant to large enterprises, cloud providers, and ISVs, basically anyone looking to optimize their data center footprint. The benefits to them are even more obvious as Levyx enables “scaling in” before “scaling out”. Why not run applications on fewer denser nodes and reduce the cost, complexities, and frustrations (performance and otherwise) of traditional scale-out infrastructures that have to date been used effectively to support Big Data applications? Levyx’s technologies can be deployed as a pluggable engine to “supercharge” database platforms and other applications that support/embed popular Key Value Store application program interfaces (APIs) from RocksDB, LevelDB, and Memcached, and soon a multitude of others. In addition, the company offers a software solution called Levyx-Spark Connector to allow Xenon to run Apache Spark deployments far more efficiently by offering in-memory speed resilient distributed datasets (RDDs) and data frames that are also persistent. With additional query pushdown features, Levyx-Spark Connector greatly accelerates operations, such as large sorts and joins; streamlines multi-job/multi-tenant workflows; and reduces the overall number of Spark nodes needed to execute those jobs. Finally, The value proposition is substantial for customers in fields such as cybersecurity (fraud or threat detection), OLTP, real-time analysis of sensor data (Internet of Things), high-speed trading, low-latency messaging, machine learning -- and for customers in sectors like Financial Services, Government & Defense, Biotech & Healthcare, eCommerce, Social Networking, Digital Advertising, and Cloud Infrastructure (including Telcos).

**Software-Defined Data Processors™**

Levyx provides a software-only, high-performance key value storage engines (Helium™) and analytics offload engine (Xenon™) that were built from scratch. The benefits of these engines hinge on multi-core optimization, query offloading, and a highly parallelized IO architecture that substitutes Flash storage for RAM while achieving equivalent or greater performance. Otherwise
put, we fully expose applications to the system in ways that take full advantage of all of the latest innovations in hardware and offload critical functions such as (storage and computations) to allow these applications to perform at their peak.

Whether we are paired with top-of-the-line high performance hardware or are running on commodity systems, Levyx allows open-source Big Data applications to operate Faster, Simpler, and Cheaper:

- Faster (by over 10X) than other in-memory key value stores because of its multi-core, flash-optimized, query pushdown, and patented indexing design.
- Simpler than other architectures, which make trade-offs between performance and storage tiering complexity -- all data is persisted by Levyx at memory speeds.
- Cheaper because Levyx substitutes random access memory (RAM) with less costly Flash storage (typically 10X cheaper per GB), yet achieves equivalent or greater performance using drastically fewer distributed commodity server nodes.

© Copyright 2013-2017 Levyx Inc. Proprietary and Confidential
Helium™ Overview

Based on our research and testing, the Helium data engine is one of the fastest KVS technologies in production today. It provides a very fast and scalable platform for storing and retrieving data items. It is specifically designed for applications that manipulate a large amount of data (billions of data items), and at the same time need to be very responsive. Real time analytics, analysis of network and server logs in data centers and cloud infrastructures, analysis of large sets of DNA sequences to find alignment and correlations, and analysis of large, multi-billion node, graphs are some of the examples of Helium applications. Helium is specifically designed to take advantage of new hardware trends, specially the emergence of CPUs with many cores; GPUs and other co-processors; and new non-volatile storage technologies, like SSDs.

Characteristics

Helium Data Store provides the best of both worlds in that it allows applications to manipulate very large amounts of data (Terabytes, scaling to Petabytes), using standard systems, but with the performance characteristics of in-memory data stores. Helium Data Store achieves this by following main design principles:

1. Scalability across multiple cores: unlike many existing data structures that are used for keeping track of data stored in a datastore, Helium is using data structures that are lock free and can scale to many cores without sacrificing performance. This allows Helium to handle processing of millions of transactions per second on a system that has many cores. Existing server processors have 48 to 96 logical cores and will have many more in the years to come;
2. Optimized for new storage and non-volatile memory technologies: Helium can store most of its data in solid-state storage (SSDs and upcoming non-volatile memory technologies);
3. Simplicity and ease of Integration; and
4. Multi-platform support.

Helium’s architecture uses the following methodologies to comply with these design principles:
Helium has the following high-level characteristics and enterprise features:

**Characteristics**
- Key/Value multi-store
- ACID compliance
- Point/range query support
- Transactions
- Compression
- Networked/linked as an embedded library
- OS and architecture agnostic
- Tunable
- Designed to index billions of objects and petabytes of data in RT
  - <100 us response, <10 us average
- Locking on object collision only
- Optimized for flash
- Reduce write/read amplification
- Direct I/O to device
- Low-level commands (e.g., trim)
- Optimized for multicore
- NUMA-aware
- Uses spinlocks
- Uses extended ISA

**Enterprise-Grade Features**
- Extensive build and test infrastructure
  - Crash recovery tests
  - Random power-off tests
  - Steady state GC tests
  - Smoke tests
  - Nightly performance and regression tests
  - JNI interface
  - CLI management tool
  - Focus on Linux/Windows/OSX
- Helium image booted into the cloud
  - Click to deploy
- Comparison w/ RocksDB: equiv. of RocksDB test (dbtest)
- Mapkeeper wrapper for YCSB benchmarking
- Customer-specific features:
  - Iterator support
  - Thin provisioning
  - Live demos, blogs, documentation, etc.

In addition, Levyx has recently added High Availability (HA) making Helium more robust and fault tolerant, and thereby suitable for mission critical environments. Helium-HA is a distributed, highly available key value store that scales the performance of Helium across many nodes. It
allows users to specify the desired level of redundancy based on their application requirements, and in case of any node failure keeps the system working.

**XENON**

**Xenon™ Overview**

Xenon is a low latency, scalable data analytics server designed to manage the retrieving, processing, and indexing of very large datasets, i.e., collections of billions of objects, spread across a tightly coupled cluster of servers, each with multi terabyte persistent storage capabilities.

Xenon is the next phase in the evolution of data processing technology as it builds on Helium’s robustness as an already-fast object store and we make it faster by getting closer to bare metal. It is applicable in distributed systems making it an excellent vehicle through which to provide enterprise features to existing Big Data platforms. Xenon uses an abstraction layer called “distributed storage class memory” that allows Big Data applications to access the data as if it is in a very large persistent, highly available, indexed, memory pool. This immediately brings high availability and persistence to large working sets that are processed by Big Data platforms like Apache Spark. For example, using Xenon, Apache Spark RDDs are persisted, highly available, and sharable among different applications — this is something that can reduce a lot of overhead and highly simplify enterprise Apache Spark deployments when multiple groups might work simultaneously on the same data set. Without Xenon, each of them would have to have their own copy and read it from HDFS or other, much slower, data sources.

Xenon is a distributed database system having the following functional capabilities:

1. Core SQL functionality: filter, projection, selection, sort, join, groupby, and aggregates on structured data, i.e., schema-based tables.
2. Support for random lookup and neighborhood search using an index rather than scan and filter.
3. Tightly integrate with the Apache Spark system for ease of deployment and use (however, fully capable to function in native mode, or serve as an off-load layer for other big-platforms and systems).
4. Scale with the number of cores in the cluster and use SSD (or other high-bandwidth, low latency persistent storage) as the storage fabric for datasets.

In practice, Xenon provides just-in-time compilation of the entire engine:

- Compiling the custom datapath, cache, and indexer from dataset schema including access to billions of random objects on 100’s of TB of Flash at sub msec access times
(enabling true OLTP) as well as sequential object manipulation on 100’s of TB of Flash at similar speeds (enabling fast OLAP); and

- Continuing to decouple data parallelism from hardware parallelism.

In effect, the user has a platform for both analytics and real-time transaction processing running on very dense nodes with in-memory performance on commodity hardware.

Characteristics

Xenon has the following characteristics:

**Characteristics**

- Core SQL functionality: filter, projection, selection, sort, join, groupby, and aggregates on structured data
- Accommodate simultaneous analytics and transactional operations on the same dataset
- Map/reduce data analytics capabilities
- Tightly integrate with the Apache Spark system
- High-performance atomic No-SQL functionality: put, get, and delete on structured or unstructured data with support for point and range query
- Scale with the number of cores in the cluster and use SSD (or other high-bandwidth, low latency persistent storage) as the storage fabric for datasets
- Provide in-memory performance
- Uses RDMA and JITC

Xenon was designed with the following key principles in mind:

1. Bypass file-system and kernel-buffers and directly perform flash friendly I/O to (a) reduce I/O latency -- important for transactional operations and (b) saturate the I/O bandwidth of
the storage devices -- important for analytics operations. Reduce read/write amplification to absolute minimum and issue I/O to the device in large sequential write and small random read patterns. Make use of NVMe technology to increase I/O efficiency.

2. Reside on a Distributed Storage Class Memory abstraction (patent pending) providing persistent byte addressable memory at flash capacities and DRAM performance across a cluster of nodes.

3. Use just-in-time compilation to perform high-bandwidth database operations as close to the metal as possible and avoid any and all form of data interpretation.

4. Utilize modern processor ISA capabilities (e.g., memory barriers, bit-level instructions, etc.) to design lock-free and scalable data-structures and algorithms.

5. Design a custom, modern, compressed, memory-resident, serializable, and lock-free indexer (patent pending) that tracks objects residing on persistent storage. Ultimately, allow high-performance in-memory point and range-query operations performed directly on the compressed indexer while being able to retrieve or update any object on persistent storage with a single read or write issue to the device.

Helium and Xenon Architectures
As depicted in the diagram above, Helium acts as storage engine distinct from but complementary to Xenon which serves as a high-performance compute engine (analytics offload engine). When data is moved to the Xenon layer, it is indexed and sorted using Helium technology for the queries that can benefit from sorted/indexed data. Most queries involve sorting, joining, or aggregating data and can benefit greatly from these super efficient, compact indices.

**Conclusion**

Now you can have *high-speed with persistence* at an affordable price.

The Holy Grail for Big Data solution providers has always been to find a way to cost-minimize hardware and supporting infrastructure in order process large amounts of data without degrading the performance (or speed) of the systems used to handle these tasks. By bridging the gap between the innovations in Big Data software (like Apache Spark, NoSQL, etc…) and the hardware innovations of the last decade (multi-core, Flash/NVMe, switches, etc…), Levyx attacks precisely those challenges and has come up with solutions that will help usher in a new era in computing. High-performance, large-scale environments are no longer niche, they are are the norm. Levyx’s data processors create in-memory-like solutions from persistent, lower-cost media and this will have dramatic impacts on end-users that require low latency access to, and fast processing of, large-scale datasets --- such as E-commerce, IoT, Cybersecurity, Social Networking, Cloud Infrastructure, Government & Defense, and Financial Services --- as these enterprises will get extreme performance at a more attractive price point than highly-customized, memory-intensive closed solutions.