WHITE PAPER

GridGain

Accelerate MySQL® for Demanding OLAP and OLTP Use Cases with Apache® Ignite™

A GridGain Systems In-Memory Computing White Paper

February 2017

© 2017 GridGain Systems, Inc. All Rights Reserved.

Contents

© 2017 GridGain Systems, Inc. All Rights Reserved.

MySQL® is a widely used, open source relational database management system (RDBMS) which is an excellent solution for many applications, including web-scale applications. However, its architecture has limitations when it comes to big data analytics. As data volumes continue to increase exponentially, businesses continue to demand accurate real-time data analysis. That is extremely difficult to achieve with MySQL alone. Many MySQL users have leveraged additional open source and commercial solutions to enhance MySQL's capabilities and add in-memory computing functionality. However, these point solutions lack the full functionality of an in-memory computing platform like Apache® Ignite™.

Apache Ignite, the leading open source in-memory computing platform, delivers both speed and scale to RDBMS-based applications. Inserted between the application and data layers and compatible with all common RDBMS, NoSQL, and Hadoop® databases, it has become a key tool for the implementation of real-time big data analytics. As an advanced high-performance cache platform for hot data, Apache Ignite can overcome the limitations of the MySQL architecture and do so better than alternative solutions such as Memcached, Redis®, Elasticsearch®, or Apache® Spark™.

Businesses typically rely on a blend of open source and commercially available technologies to efficiently store and process large data volumes. System architects pursue the most optimal available solutions and then strive to make them work together. The Apache Ignite in-memory computing platform can easily be inserted into existing MySQL architectures to overcome the inherent limitations. This results in dramatically faster large-scale analytics and transactions without significantly increasing development and management overhead.

Five Limitations of MySQL

A close look at the strengths and weaknesses of MySQL reveals several use cases where the RDBMS, powerful though it is, can benefit from some assistance:

Delivering Hot Data

In large applications, the data cache stored in RAM can grow very large and be subjected to thousands or even millions of requests per second. MySQL does not have a strong memory-focused search engine. Because it is not designed for very high concurrency, users can be exposed to bottlenecks and periodic performance issues. MySQL is saddled with relatively high overhead and cannot deliver optimal speed.

A Solution: It is common to use either Memcached or Redis as external solutions for caches and other hot data needs, to eliminate the overhead of SQL parsing and transactions. But caching is difficult because you risk reading data that is not current. It is also possible to use internal scalability improvements in MySQL, such as the thread pool, available either as a MySQL enterprise feature or as an open source implementation. But such features typically help many queries to run concurrently, without improving the performance of any given query.

A Better Solution: The Apache Ignite in-memory computing platform includes an in-memory data grid and an in-memory SQL grid. Redis and Memcached are primarily in-memory key-value stores. All three

© 2017 GridGain Systems, Inc. All Rights Reserved.

solutions partition and cache data in memory and they can be scaled out across distributed clusters. However, there are many differences in the way caching, transactions, persistence, and data querying are supported. The Apache Ignite in-memory computing platform includes many additional features not in Redis or Memcached that are often highly valuable for companies moving to in-memory computing. For example, ANSI-99 SQL, distributed ACID-compliant transactions, advanced computing and streaming capabilities, and more.

Dealing with Highly Volatile Data

When thousands of updates per second are applied to a single database row (for example, flash online sales for high-demand concert tickets), it is crucial to maintain exact values at every second. MySQL is designed around full transactional semantics with support for long transactions and works with disk-based log durability. It is therefore not well suited for use with this kind of highly volatile data.

A Solution: To some extent, good data design within MySQL can address this issue. Splitting the counter over several rows can help, and optimal configuration of the MySQL installation can yield up to ten times better performance than a stock MySQL configuration. Parallel replication, another historical problem, has been addressed in MySQL 5.7, or by solutions such as Percona XtraDB Cluster. Some users even move frequently-updated data to Memcached or Redis, and then periodically synchronize to the RDBMS.

A Better Solution: Apache Ignite automatically distributes data across all nodes in a cluster. Replication between nodes and clusters is configurable and takes place automatically. Apache Ignite can be configured to provide the needed level of consistency. Once the data is in Apache Ignite, you can leverage the ANSI-99 distributed SQL engine and ACID compliant transactions in order to work with your data in the same way you work with it in MySQL. The Apache Ignite in-memory computing platform also includes many additional features not included in Percona XtraDB Cluster, Redis or Memcached that are often highly valuable for companies that are moving to in-memory computing.

Handling Large Data Volumes

MySQL was originally designed as a single-node system and not with the modern data center concept in mind. Today's largest MySQL installations cannot scale by using MySQL as a single system and must rely on sharding, or splitting a data set over multiple nodes or instances. However, most sharding solutions in MySQL are manual and make application code more complex. Any performance gain is lost when queries must access data across multiple shards.

A Solution: Tools such as Vitess, a framework YouTube released for MySQL sharding, or ProxySQL are often used to help implement sharding. Memcached and Redis also offer front-end solutions, as do MongoDB and Cassandra.

A Better Solution: Apache Ignite was built from the ground up as a high performance and highly scalable distributed in-memory computing platform. There are no limitations to the amount of CPU and memory that can be used by any node. Furthermore, nodes are automatically load balanced. Data is automatically distributed across all nodes in a cluster so manual sharding is not necessary. Apache Ignite

© 2017 GridGain Systems, Inc. All Rights Reserved.

is a complete in-memory computing platform that includes many additional features beyond those offered by point solutions such as Vitess, Redis, Memcached, MongoDB, and Cassandra.

Providing Analytics

MySQL was not designed for running complicated queries against massive data volumes which requires crunching through a lot of data on a huge scale. MySQL optimizer is quite limited, executing a single query at a time using a single thread. A given MySQL query can neither scale among multiple CPU cores in a single system nor execute distributed queries across multiple nodes.

A Solution: Although MySQL offers no comprehensive solution for robust, large-scale data processing, many external solutions, such as Hadoop® and Apache® Spark™, are often used. Vertica and ClickHouse have also emerged as worthy analytics solutions.

A Better Solution: Apache Ignite easily integrates with Hadoop and Spark, using in-memory technology to complement these technologies and achieve significantly better performance and scale. The Apache Ignite in-memory SQL grid is highly optimized and easily tuned to execute high performance SQL queries. The in-memory SQL grid provides JDBC/ODBC connections to any analytics solution.

Powering Full Text Searches

MySQL can handle basic full text searches. However, because of its inability to manage parallel processing, searches do not scale well as data volumes increase.

A Solution: For small-scale search applications, InnoDB, first available with MySQL 5.6, can help. When the search spans beyond a few gigabytes, however, an advanced search solution such as Elasticsearch, Apache Solr, Crate.io or Sphinx Search makes sense.

A Better Solution: Apache Ignite supports text-based queries based on Lucene indexing. Text searches are distributed across the cluster for parallel processing against data stored in memory. This results in high performance and highly scalable full text searches.

When Two Trends Converge

Although MySQL is an excellent choice for many applications, two major trends have converged to change the landscape of large-scale data processing. Together they are forcing a wholesale change in the database world.

1. **In-memory computing** and the concept of caching has been around for 40 years, but it has come to the forefront as real-time data processing has become standard operating procedure (or at least the goal) of enterprises of all sizes. In-memory computing is the key to that speed, and at some point in the not-too-distant future, advances in memory technology may lead us to a world of in-memory storage as well, with disks serving only as historical backups.

© 2017 GridGain Systems, Inc. All Rights Reserved.

2. **Memory prices** have dropped to the point where, for example, a clustered terabyte of RAM costs about $20,000, making it affordable for large enterprises that demand real-time data processing and analytics of data sets of that scale.

The good news is that as these trends converge, in-memory platforms are ready to deliver as the most sophisticated distributed systems in existence because they are efficient and well-architected. The largest implementations of Apache Ignite rival the largest installations of Cassandra and Hadoop with several thousand nodes and clusters – while providing greater performance and scalability.

The Speed and Power of Apache Ignite

Apache Ignite is a very powerful option for accelerating MySQL and other RDBMS-based applications. Apache Ignite is deployed as a layer that slides architecturally between data sources and applications without forcing users to keep their entire data set in RAM.

Apache Ignite provides speed in two important ways. It keeps data in RAM in the in-memory data grid, distributed across a cluster of servers. By caching the data in memory, the system avoids the delays inherent in retrieving data from disk prior to processing. The system then leverages its in-memory compute grid to parallel process queries. Through parallel processing, the system can leverage the processing power of multiple nodes in the cluster and calculate results faster. Caching and parallel processing the data in memory improves processing speeds by 1,000x or more versus disk-based databases like MySQL which were designed to run on a single node.

Apache Ignite provides scalability by allowing the size of the in-memory grid to be increased simply by adding nodes to the cluster. As new nodes are added, data is distributed across the larger grid and the new node's computing power is added to the in-memory compute grid.

Apache Ignite offers an ANSI-99 SQL compliant in-memory SQL grid which supports standard SQL DDL commands via an ODBC/JDBC API. MySQL users can continue to use SQL or any of a number of common, supported protocols to access and process their data in Apache Ignite.

The Apache Ignite in-memory computing platform has many features and components. Its modules include:

- **In-Memory Data Grid**. The In-Memory Data Grid is horizontally scalable with the ability to add nodes on demand in real-time. It can linearly scale to hundreds of nodes with strong semantics for data locality and affinity data routing to reduce redundant data noise. The In-Memory Data Grid supports local, replicated, and partitioned data sets.

- **In-Memory SQL Grid**. The In-Memory SQL Grid provides in-memory distributed database capabilities. The In-Memory SQL Grid is horizontally scalable, fault tolerant and ANSI SQL-99 compliant. It fully supports all SQL and DML commands. The In-Memory SQL Grid allows you to interact with the Apache Ignite platform using standard SQL language through JDBC or ODBC APIs without custom coding.

© 2017 GridGain Systems, Inc. All Rights Reserved.

- **In-Memory Compute Grid**. Beyond simply distributing data, Apache Ignite provides the ability to work with the data in place. The In-Memory Compute Grid lets it send computations, queries, and entry processing to where the data resides.

- **In-Memory Streaming/CEP**. The In-Memory Streaming feature uses programmatic coding with rich data-indexing support to provide complex event processing (CEP) capabilities on streaming data. Apache Ignite also provides comprehensive support for customizable event workflows and windowing.

- **In-Memory File System**. The Apache Ignite acceleration feature for Hadoop is a high-performance implementation of a job tracker that replaces the standard Hadoop MapReduce and HDFS with the compliant Ignite In-Memory File System (IGFS). IGFS works in dual mode, either in tandem with HDFS or as a standalone primary file system in the Hadoop cluster. In the latter case, it serves as an intelligent caching layer with HDFS configured as the primary file system.

- **In-Memory Service Grid**. The In-Memory Service Grid provides complete control over services being deployed on the cluster. It lets you control how many instances of your service should be deployed on each cluster node, ensuring proper deployment and fault tolerance.

- **Apache Spark integration.** Apache Ignite provides an implementation of the Spark RDD abstraction that lets you easily share state in-memory across Spark jobs. You can run faster SQL queries against the Spark data because Apache Ignite indexes the data set while Spark runs a full scan of the data for every query. In addition, Apache Ignite powers mutable RDDs through its in-memory data storage.

- **Advanced Clustering**. Apache Ignite provides one of the most sophisticated clustering technologies on Java Virtual Machines (JVMs). The ability to connect and manage a heterogeneous set of computing devices is at the core of the Apache Ignite distributed processing capabilities. Clustering capabilities are fully exposed to the end user, and developers have full control with many advanced features.

Apache Ignite Tames a Raging River of Data

In-memory computing helps meet the needs of businesses drowning in data. For example, Apache Ignite is used by a company that develops an Internet of Things (IoT) software platform to address a large, real-time data processing challenge. The company's connected energy meters deliver data from millions of homes, and the company forecasted that its existing database infrastructure would soon no longer be able to keep up. Not only did it need to ingest massive amounts of data constantly, it also had to overlay current electricity rates and perform compute-heavy calculations on the fly in order to produce regular analytical reports combining current state and recent history.

The company was committed to keeping as much of its database infrastructure as possible, so the solution was to add the Apache Ignite in-memory computing platform to their architecture. This

© 2017 GridGain Systems, Inc. All Rights Reserved.

provides an in-memory computing platform in which to perform the required real-time data processing and analytics and as much scalability as the company needs — up to thousands of nodes supporting terabytes of data, as necessary — as the needs of each of its customers grow.

Today the platform can store a week of current data entirely in RAM, vastly improving the system's ability to deal with the never-ending rush of data it must process in real-time.

Focus on Speed and Scalability

Databases such as MySQL have always been hard to scale but today's demands for real-time processing of large data volumes require solutions to that challenge. While MySQL is an excellent solution for many applications, it is not designed for fast parallel processing and endless expansion. Many MySQL users have leveraged additional open source and commercial solutions to enhance MySQL's capabilities and add in-memory computing functionality, but these point solutions lack the full-functionality of an in-memory computing platform like Apache Ignite. Apache Ignite can provide massive gains in speed and scalability. Adding an in-memory computing layer provides users with the best of both worlds: the opportunity to maintain their current databases and application code with minimal or no changes while gaining vastly improved performance and scalability.

The open source ecosystem is always evolving. It is crucial to be on the lookout for new and innovative solutions that enable development teams to deliver greater operational efficiency and easy-to-manage analytical tools. Apache Ignite is a prime example of that kind of straightforward solution. This makes Apache Ignite well-suited to help eliminate the pain points of MySQL.

Contact GridGain Systems

To learn more about how GridGain In-Memory Data Fabric can help your business, please email our sales team at sales@gridgain.com, call us at +1 (650) 241-2281 (US) or +44 (0) 7775 835 770 (Europe), or complete our contact form to have us contact you.

About GridGain Systems

GridGain Systems is revolutionizing real-time data access and processing by offering enterprise-grade in-memory computing solutions built on Apache® Ignite™. GridGain solutions are used by global enterprises in financial, software, ecommerce, retail, online business services, healthcare, telecom and other major sectors. GridGain solutions connect data stores (SQL, NoSQL, and Apache™ Hadoop®) with cloud-scale applications and enable massive data throughput and ultra-low latencies across a scalable, distributed cluster of commodity servers. GridGain is the most comprehensive, enterprise-grade in-memory computing platform for high volume ACID transactions, real-time analytics, and hybrid transactional/analytical processing. For more information, visit gridgain.com.

© 2017 GridGain Systems, Inc. All Rights Reserved.

COPYRIGHT AND TRADEMARK INFORMATION

© 2017 GridGain Systems. All rights reserved. This document is provided "as is". Information and views expressed in this document, including URL and other web site references, may change without notice. This document does not provide you with any legal rights to any intellectual property in any GridGain product. You may copy and use this document for your internal reference purposes. GridGain is a trademark or registered trademark of GridGain Systems, Inc. Windows, .NET and C# are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. JEE and Java are either registered trademarks or trademarks of SUN Microsystems and/or Oracle Corporation in the United States and/or other countries. Apache, Apache Ignite, Ignite, the Apache Ignite logo, Apache Spark, Spark, Apache Hadoop, Hadoop, Apache Cassandra, and Cassandra are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. All other trademarks and trade names are the property of their respective owners and used here for identification purposes only.

© 2017 GridGain Systems, Inc. All Rights Reserved.