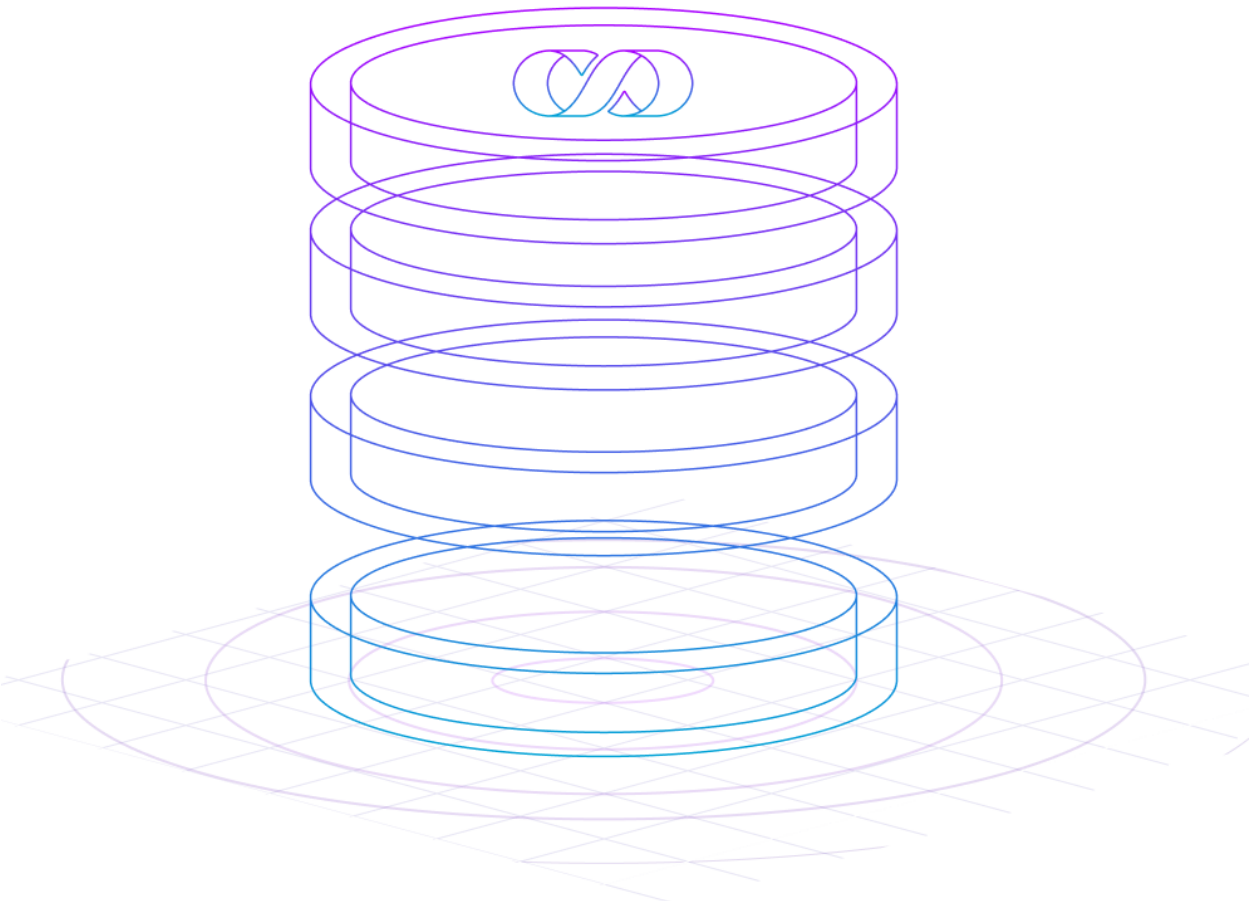
IN DEPTH COMPARISON

Modernizing Data Platforms with MemSQL



Written by

Nithin Krishna Reghunathan, Technical Evangelist

Krishna Manoharan, Director of Solution Architecture

memSQL

2019

Table of Contents

1. Business and Technology Drivers for Modernizing Data Platforms

1.1 - A brief history of data platforms

Historically, databases fit into one of two categories: those optimized for online transactional processing (OLTP) and those optimized for online analytical processing (OLAP).

I - OLTP

An OLTP-optimized database is essentially a database of record, serving as a single source of truth for downstream systems. OLTP databases frequently power large transactional workloads such as enterprise resource planning (ERP), retail sales transactions, and inventory management.

A transaction operates on a small set of records at any point in time. OLTP systems are designed to efficiently process transactions. OLTP systems manage multiple such transactions concurrently.

To summarize, OLTP systems are designed and optimized for highly concurrent, transactional workloads.

II - OLAP

OLAP systems allow you to generate actionable insights from analyzing data across lines of business in an enterprise.

OLAP runs on top of a data warehouse system. OLAP systems are designed to efficiently ingest data in bulk and query large sets of data. These databases use storage formats that quickly perform operations needed for analytics, such as table scans and aggregations.

To summarize, OLAP systems are designed and optimized for low-concurrency, batch-oriented analytical workloads.

Depending on the size and complexity of the data warehouse, companies may decide to split the data warehouse into multiple data marts that each house a relevant subset of the data.

III - Traditional usage patterns of OLTP and OLAP data platforms

Transactional systems are traditionally separate from data warehouse systems, even though they could potentially be combined into a single system. Transaction systems are often revenue generating, have strict availability requirements, and are viewed as mission critical.

Data warehouse systems are used to analyze large amounts of data and require a lot of computing resources. They are not generally as mission critical.

Combining data warehouse and transaction systems in a single database generally results in the transaction workload suffering and thus affecting business adversely. Hence, separation of the two has become standard.

At scheduled intervals, the data is copied to an OLAP-optimized data warehouse from multiple transactions system for analysis and archival requirements. A data warehouse is batch-oriented and not real-time.

Additionally, there is often a third component, called an operational data store, that is used to support operational analytics. This allows business to have near real-time visibility into rapidly changing events, such as orders and/or customer interaction.

Success of the operational data store is driven by the ability to handle streaming ingest of data with concurrent analytical queries.

An operational data store receives transactions from an OLTP system in a minimally intrusive manner using techniques such as change data capture. It also serves as a source for the data warehouse.

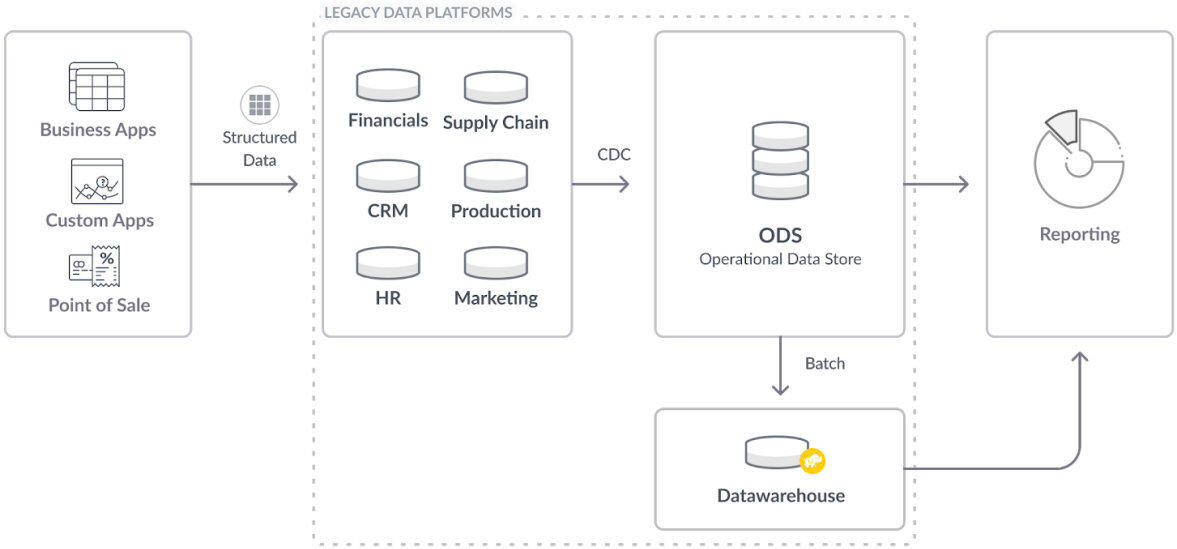


Figure 1. Legacy multi-tiered architecture

1.2 Challenges with legacy data platforms

In all, this legacy architecture with separate transactional, operational, and data warehouse requirements results in data duplication, complexity, and maintenance overhead. This model is inflexible and unable to adapt to changes in business practices.

Legacy platforms are also not suited to meet modern requirements for businesses that are on the path to digital transformation.

A key component of digital transformation is data-driven decision making. With the explosion of data from multiple sources and different formats, consuming, understanding,

and managing this data is critical to the success of a modern enterprise. Data is the ultimate asset, and using data to generate actionable insights to drive revenue growth and reduce cost and risk is the goal for a modern enterprise.

With the growth in mobile phones, smart devices, and social media, understanding how users interact online is another important component of being data-driven. This has resulted in the growth of system of engagement (SOE) platforms, which are designed to improve enterprise user experience and retain users. In many ways, SOE is the modern ERP.

These sources deliver a combination of structured and semi-structured data. Legacy data platforms support these data formats, but they are unable to consume and report on semi-structured data efficiently.

There is a growing need to use this data to be proactive rather than reactive. Advances in artificial intelligence (AI) and machine learning (ML) now make this possible. As data is being ingested, it can be used to train models and score incoming data to help predict future trends.

Legacy platforms are not suited to support AI/ML due to inherent structural and performance limitations.

Businesses that are proactive need to adapt to change rapidly. Deploying data platforms in the cloud has enabled business to manage cost and be flexible in terms of deployment options.

Legacy platforms are by and large not cloud-friendly.

In summary, legacy data platforms are ill-suited to meet modern enterprise requirements. They are unable to:

1. Manage both structured and semi/unstructured data sources natively and seamlessly with the familiarity of SQL.

2. Meet the performance requirements to manage the explosion of data and user concurrency.
3. Provide a path to a cloud-native architecture for cost management, agility, and flexibility.
4. Enable a continuous learning model wherein a business can learn and adapt as data changes.
5. Enable converged data processing for new apps and analytics.
6. Reduce cost, complexity, and maintenance overhead.

1.3 - MemSQL as a modern data platform

In order to flourish in today's hyper-competitive environment, enterprises requires a data platform that is designed to meet and exceed today's demanding requirements.

I - Modern architecture

First released in 2011, MemSQL is a third generation RDBMS written in C/C++. MemSQL was designed to run efficiently on modern systems; both multi-core systems with a big memory footprint and lower-powered edge computing devices.

MemSQL natively supports structured, semi-structured, and unstructured (full-text search) data. This allows support for legacy transactions and/or analytics, but also the ability to manage data from sources such as mobile phones, social media, and smart devices on a single platform. MemSQL is ANSI SQL-compatible and so allows for easy management of data and robust analytics support.

MemSQL has built-in connectors to Kafka, Spark, S3, and Hadoop, as well as legacy transactional systems. This allows MemSQL to integrate with an existing ecosystem on the path to modernization.

II - Highly performant and scalable

MemSQL incorporates recent developments in lock-free/non-blocking algorithms and multi-version concurrency control (MVCC). This allows MemSQL to ingest data and run analytics on a single platform, without compromising on performance, and while eliminating legacy platform issues with data duplication, complexity, and high maintenance overhead. MemSQL compiles queries to machine code to take advantage of modern chip capabilities, such as single instruction, multiple data (SIMD) instructions. SIMD and query vectorization delivers extremely fast processing by leveraging AVX-2/512 on-chip caching. MemSQL, being a distributed platform, allows you to scale out horizontally or vertically as needed. MemSQL can be deployed on a single system or on multiple systems easily.

III - Flexible deployment options, elastic and cloud-native

Being cloud-native, MemSQL runs on virtual machines, containers, and physical servers. This provides flexibility and easy cost management. MemSQL runs on both private and public clouds and is a natural fit for Kubernetes. MemSQL is one of the first major databases to support a Kubernetes Operator, currently in beta format for both open source Kubernetes and for Kubernetes in the Red Hat OpenShift environment.

IV - Native integration with AI/ML

MemSQL allows you to integrate with existing libraries such as SparkML and TensorFlow, or to use native and user-defined MemSQL ML functions to create and train models for predictive analytics. The functionality is included within the database and is not a separate component with an additional license and associated costs. As an integrated solution, developers can take advantage of the performance of MemSQL in building,

training, and deploying models quickly. Since MemSQL supports streaming ingest, it allows for real-time scoring as well.

V - Ease of management

MemSQL is easy to install and maintain. Typical installation of a distributed cluster takes less than 30 minutes. A single system install takes approximately 5 minutes or less. Installations are easy to automate. This reduces the requirement to have a dedicated team to install and maintain MemSQL.

MemSQL is MySQL wire protocol-compliant and allows easy integration into existing ecosystems. The default configuration works out of the box for most workloads and runs on most Linux variants.

VI - Cost-effective

With MemSQL, all enterprise features, such as partitioning, security, high availability (HA), and disaster recovery (DR), are included in the product and not licensed separately. Ability to run on the cloud or on-premises, including in containers and virtual machines, allows for deployment flexibility. MemSQL, being modern and efficient, allows workloads to run better on lesser hardware when compared to legacy systems. Thus cost savings are realized in reduced software spend, cost of deployment, and maintenance costs in comparison to legacy vendors. The advantage in total cost of ownership varies depending on the legacy vendor under consideration.

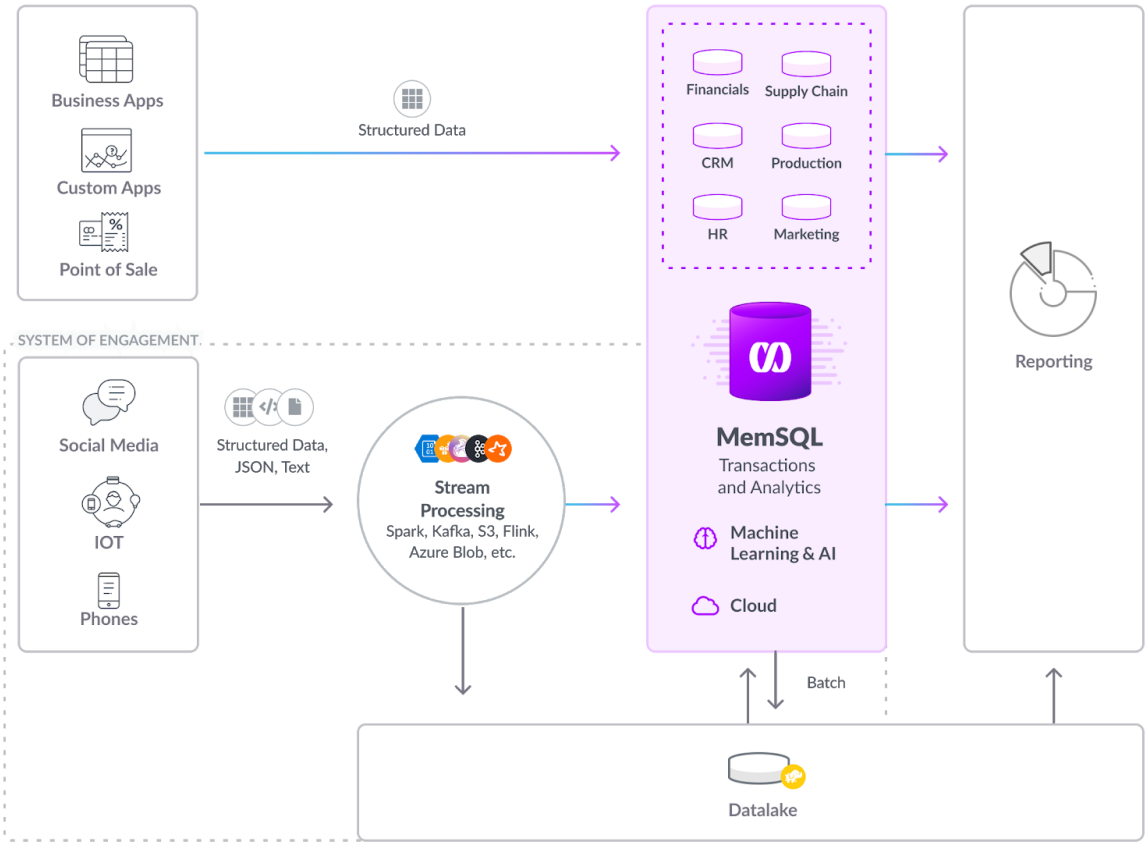


Figure 2. Modernizing legacy data platforms with MemSQL

1.4 - Comparing Oracle to MemSQL

I - Legacy architecture

Oracle Database was first released in 1979 and is currently in its nineteenth version. Oracle Database can be considered a leader in the relational database market with respect to structured data. However, it is not designed or able to manage semi-structured and unstructured data well enough to deliver on today's requirements, such as systems of engagement.

Traditional Oracle databases do not support mixed workloads (such as concurrent transactions and analytics) well. As a result, most deployments of Oracle in the

enterprise space have a clear separation between transactions and analytics. This separation adds to the overall cost and complexity of Oracle data platforms.

Oracle also does not integrate with data sources such as Kafka, S3, or Hadoop as an out-of-the-box function. Most such functionality is licensed separately. This results in additional cost and time delays to integrate Oracle within any ecosystem that has both transactional and modern data sources.

II - Performance and scalability limitations

Oracle was designed and developed on the computers of the 1970s and 1980s - standalone, single-processor systems with limited memory. As a result, Oracle is not built from the ground up as a scale-out database.

Oracle has since released a scale-out version of the database with a feature called Real Application Clusters (RAC). RAC uses a shared storage model, which limits horizontal scale-out. Instead, vertical scale-out tends to be the predominant model. Vertical scale-out has several limitations. Database software and the operating system have to scale to take advantage of the increased CPU and memory on the system. Tall vertical systems needed for large and demanding workloads are also non-standard hardware models requiring additional data center considerations.

Oracle has released an appliance, Exadata, to address some of the scalability and performance issues that are not met by RAC. However, this is highly expensive, and adds complexity and maintenance challenges to the overall platform. And because Exadata is an appliance, it is proprietary as well.

Oracle has released an in-memory option with 12C (IMDB) and higher versions in order to provide capabilities including columnar performance on queries and address some of Oracle's limitations on analytics. This option is an add-on and licensed separately. Being a bolt-on, it has several restrictions on the size of the data, and it requires maintaining multiple copies of data in synchronization.

III - Not cloud-friendly nor elastic

With regards to deploying in the cloud, Oracle has limited options with RAC deployments (only supported on Oracle Cloud). Single-instance databases are otherwise supported in the cloud.

Oracle RAC is not elastic-friendly by design. Most enterprise Oracle environments are an always-on cost.

IV - AI/ML capabilities as an additional add-on and licensed separately

AI/ML capabilities are available as part of Oracle Data Mining, included with the Oracle Advanced Analytics Package. Oracle Data Mining allows customers to build, train and deploy models.

This is a separate licensed component and requires Oracle Enterprise Edition.

V - Management and support challenges

Oracle has generally required a dedicated DBA team to install, maintain, and manage. It is not DevOps-friendly. A typical RAC install requires a team of system, storage, and database administrators to setup the environment prior to install. An install process can take several hours depending on configuration. Exadata systems are even harder and more expensive to set up and manage.

Single-instance Oracle systems are easier to install; however, due to the complexity of Oracle, even single-instance systems require dedicated and well-trained resources to manage them.

Performance optimization and troubleshooting require specialized skills that tend to be expensive, especiallyexpensive especially for complex workloads.

VI - Cost inefficiencies

With Oracle, enterprise features are licensed separately. The Oracle licensing model can be considered restrictive, wherein an end user is required to track which features are used, and by whom, in order to comply with the licensing agreement. Oracle licensing audits are a tactic used by Oracle to pressurize users to comply with licensing restrictions or pay for additional licensing options.

Scalability requires purchasing Oracle RAC licenses or upgrading to Exadata. RAC workloads require hardware to be Oracle-compliant, as these workloads use shared storage (generally a storage area network) and require specialized skills to install and manage.

2. Benefits of MemSQL over Oracle

Fundamentals	Oracle	MemSQL	
ARCHITECTURE	Limited horizontal and vertical scaling and requires specialized HW	Native horizontal and vertical scaling on commodity HW	*Simplicity, cost savings, flexibility, scalability*
DATABASE ENGINE	First generation RDBMS designed for OLTP now retrofitted for analytics	Third generation RDBMS designed for transactions and analytics from the ground up.	*Out of the box performance capabilities*
	Designed primarily for structured data	Natively supports structured data, semi-structured data, and full-text search	
	Default rowstore (on-disk compression licensed separately)	Rowstore (in-memory compression)	*All features included in the base product*
	Columnstore is licensed separately and on Oracle Storage only	Columnstore as part of the base product	
CLOUD READY	RAC requires to use Oracle Cloud only	Runs on physical machines and in the cloud, in containers or VMs, or in a Kubernetes stack	*Easy path to modernize legacy infrastructure*
	Single instances can run on physical machines and private/public cloud	Easy lift and shift to private/public cloud	*Cost savings, flexibility*
AI/ML READY	Licensed separately Add-on option - available with Oracle Advanced Analytics	Native AI/ML in the engine, including support for vector embeddings from deep learning	*In-DB AI pattern recognition* *Learn and adapt as your data changes*

INTEGRATIONS	Lacks integration with newer data sources	Native integration with legacy RDBMS and newer data sources (S3, Kafka, HDFS, text)	*Seamless integration & flexibility*

Performance	Oracle	MemSQL	
QUERY PERFORMANCE	Auto and manual parallelism	Auto and manual parallelism	*Better performance out of the box*
	SIMD requires IMDB	Utilizes SIMD and vectorization	*Takes advantage of native chip capabilities*
	No query compilation	JIT-compiled queries	*Minimal performance optimization needed*
	Undo and rollback segments Heavy use of latches and enqueues (locks)	Lock-free structures and modern MVCC	

Cluster Operations Management	Oracle	MemSQL	
INSTALL	Complex install process requiring multiple teams to coordinate activities.	Fully automated install process. Fairly simple to deploy big clusters.	*Saves time, cost. Results in faster go-live*
UPGRADES/PATC	Complex multi-step	Easy one-step process.	*Saves time, cost, and*

HING	process. Can be error-prone.	Completely automated.	*reduces management overhead. Predictable outage windows for businesses.*
DBA & DEVELOPER	N/A	Existing Oracle DBA/Developer can easily pick up MemSQL	*Saves time and cost*
BACKUP/ RESTORES	Complex process	Fully distributed, self contained High performance Backup to cloud object store	*Saves time and reduces management for backup to cloud object store*
SECURITY	Robust security model	Robust security model	*Comparable to Oracle*
APPLICATION SUPPORT	Oracle drivers	MySQL wire protocol-compliant	*Easy integration with any app that supports MySQL wire protocol*

COST SAVINGS	Oracle	MemSQL	
HW REQUIREMENTS	Proprietary HW or Oracle Cloud only	Runs anywhere - commodity HW, virtual machines, public or private cloud, edge devices	*Highly flexible deployment, cloud neutral*
DEVELOPMENT & MIGRATION EFFORT	N/A	Run Oracle queries as-is MPSQL very similar to PL SQL	*No code rewrite needed for front end. Saves time and effort. Quicker go-live.*
LICENSING	Complex Licensing. Enterprise features are licensed by component. DR is a separate license cost.	Node-based licensing - no hidden costs. All features included with the license. DR is a built-in feature.	*Honest and upfront pricing.*

3. Migration from Oracle to MemSQL

Both Oracle and MemSQL support ANSI SQL and have similar data types. MemSQL also supports many commonly used Oracle inbuilt functions, and MemSQL MPSQL is similar to Oracle PL/SQL. This allows for quick migration from Oracle to MemSQL.

Data and DDL migration can be easily automated using various commercial or open source tools. Stored procedures require manual conversion.

Some of the tools for data migration include:

- MemSQL in-built replication service (currently in beta)
- Attunity replication service (certified by MemSQL)
- Striim
- Streamsets
- Spark

4. Summary and Conclusion

Summary	Oracle	MemSQL
FOUNDATION	Designed primarily for transactions Limited cloud support AI/ML as add-ons Integration with some modern sources as add-ons Structured, semi-structured and full text on unstructured	Transactions and/or analytics Cloud-ready AI/ML ready Native integration with legacy and modern sources Structured, semi-structured and full text on unstructured
PERFORMANCE & SCALABILITY	Single and limited distributed (RAC) Extensive use of latches/enqueues for locking Limited query performance	Single or distributed Lock-free structures Advanced query execution
DEPLOYMENT	Distributed (RAC) only on Oracle Cloud Single on physical, VM, cloud	Commodity HW, VM, Cloud Kubernetes
MANAGEMENT	Requires dedicated team to manage	Designed for DevOps Easy transition from Oracle to MemSQL
COST	Most enterprise features are add-on Likes to audit customers frequently	All inclusive simple licensing Easy migration from legacy

4.2 - Conclusion

Modern enterprises requires a data platform that is versatile, cost efficient, and performant. Not only does the data platform need to support and improve legacy workloads, but also be able to deliver on new business requirements.

MemSQL is a modern data platform that is well suited to meet today's demanding requirements. It offers an easy migration path from legacy platforms, is cloud-friendly, and supports modern workloads seamlessly.

MemSQL allows for infrastructure convergence, simplicity, and support for predictive capabilities in a cost-effective and highly performant manner.

5. Success stories

5.1 - Global bank replaced multi-tiered Oracle RAC with MemSQL

A global bank had added several layers to its legacy environment for storing customer data. Its traditional Oracle database was migrated to RAC for added scale. Analytic

queries were beginning to slow as data grew, prompting the team to add a cache and a NoSQL database layer to support the rise in user concurrency driven by mobile app access. This bank was able to replace all three layers with a single MemSQL cluster. The firm was able to support higher user concurrency with low-latency queries using standard cloud hardware utilizing a modern, scale-out architecture.

5.2 - Energy sector giant replaced Oracle with MemSQL

One of the world's largest energy companies was driven to replace Oracle with MemSQL to improve financial forecasting and profitability analysis. This customer saw 20x performance improvement over Oracle on forecasting. This means that company executives can discuss fresh results during exec meetings instead of discussing stale reports.

Financial summary reports that previously took four hours and 12 hours, respectively, now take less than a minute in MemSQL. This turned the review of company performance metrics from a periodic task into a 24/7 operation, available to any executive at any time. These reports are now also available to hundreds of concurrent users, which was previously impossible on the legacy Oracle platform.

This customer first migrated the analytic applications to MemSQL by using Change Data Capture functions from Oracle RAC. The first phase was delivered within three months, significantly de-risking the rollout of this critical business function.

5.3 - Global fixed-income manager replaced Oracle with MemSQL

A global fixed-income manager chose MemSQL over Oracle In-Memory for its asset manager dashboards. This asset manager built its portfolio management dashboard on MemSQL in part due to response time, but also due to the operational flexibility. The asset managers are now able to quickly access information about their portfolios and

update their positions with changes in the market, allowing the firm to better serve their investors. The need to support several hundred users concurrently, to promote an elastic architecture to handle seasonality, and to grow without large upfront capital expenses were all key drivers.

5.4 - Large CDN replaced Oracle with MemSQL

A large content delivery network (CDN) and cloud services provider re-platformed its global billing platform from Oracle to MemSQL. In moving from Oracle, this firm was able to enhance billing speed and efficiency for thousands of customers around the world. Ingest throughput was accelerated from hundreds of thousands of events to millions per second. They also accelerated query performance, which allowed them to move from monthly billing to daily billing cycles, helping avoid the loss of millions of dollars in annual recurring revenues. Not only was the new configuration faster, by orders of magnitude, it was also one-fifth the cost of the maintenance Oracle contract.

For more information, or to talk to a MemSQL specialist about your database migration, please contact us at info@memsql.com