



# Optimize Oracle Databases with Violin flash Memory Arrays

Run Oracle databases at the speed of memory at a higher transactional throughput and lower total cost of ownership with Violin flash Memory Arrays

## Highlights

### Accelerated Throughput

- 20x faster application response times
- Higher number of Oracle transactions at sustained performance
- Predictable and consistent application performance

### Resilient Database Performance

- Consistent low latency at any scale and any workload
- Sustained maximum performance at connectivity bandwidth levels
- Massively scalable architecture allowing for easily scalable I/O without upgrades in hardware

### Lower Total Cost of Ownership

- Decrease power, cooling and space costs by up to 80%
- Reduce cost per transaction by reducing cores, spindles and ports, as well as Oracle licenses
- Better utilize existing storage

## Business Needs

Today's enterprises are under constant pressure to reduce costs, improve performance, and increase revenue. The onus is on IT to support the rapid growth in data, along with real-time processing demands and detailed analytics requests, all while reducing operational costs.

To keep pace with these rapidly changing business needs, IT infrastructure has transformed, leveraging faster networks, multi-core CPUs, and server virtualization technologies to scale operations. Storage, however, has not kept pace. The legacy technology of spinning disks has not improved in line with these business demands and its I/O limitations are heavily impacting business critical operations, reducing the overall efficiency and performance of Oracle databases and applications.

## Business Challenges

Storage acts as the ultimate inhibitor in database performance. The time needed to read or write data from storage directly impacts database performance. Current hard disk media is comprised of legacy technology with moving parts and spinning platters, unable to keep up with Oracle's multi-tenanted workloads. This leads to slow response times (latency), and low throughput (IOPs).

When data is requested, the database must read from the storage medium, transfer responses across the network, and copy them into the database memory buffer. The requesting process must wait until this operation is complete, and any other process requiring the same data must also wait. On average, disk response times are between 3ms to 7ms, but during heavy periods can increase to 15-20ms. Waiting on I/O negatively impacts CPU efficiency, leading to wasted cycles and under-utilization of resources, and translating into revenue and productivity loss as well as customer dissatisfaction.

## The Violin Difference

Violin flash Memory Arrays address the I/O limitations of traditional storage solutions by storing all the active data in flash memory and providing sustained peak performance. Whether you use Oracle database software for data warehousing, online transaction processing (OLTP), or analytics, Violin Arrays ensure the lowest latency for data access, provide high bandwidth, and scale to hundreds of terabytes of data so that even the largest Oracle databases can be stored in flash memory. Violin flash Memory Arrays provide:

- Faster transaction and query performance
- Higher CPU utilization
- Up to 80% lower costs for power, cooling and space
- Fewer database licenses

## Unleash Your Applications with the Flash-Enabled Oracle Database

Since the early days of the Oracle Database product, DBAs have developed methodologies to avoid accessing storage in order to improve database performance. Some of these include:

- Using database buffer caches and buffer cache hit ratios to pin data blocks in memory where possible
- Fine-tuning PGA memory components for the least use of temporary tablespaces
- Creating multiple complex indexes to avoid the dreaded penalty of the “full table scan”
- Gathering, locking or even manually altering object and workload statistics to influence optimizer behavior
- Embedding application SQL with hints or overriding execution plans with profiles and outlines to ensure a specific plan will be used

The penalty for using the wrong access method on disk storage is extreme and costly: reading a data block from a disk array can take a million times longer than reading the same data block from memory. Systems running on disk are inherently unbalanced.

Violin flash Memory Arrays address this imbalance by allowing you to run your entire database in flash memory. Violin uses the latest in flash technology to provide higher performance with consistent low-latency, higher IOPs, and higher throughput for Oracle database environments. By reading and writing data blocks within microseconds, Violin eliminates I/O contention and improves productivity by making applications work harder.

With a flash-enabled database, the overhead of accessing storage is far less significant, resulting in extreme performance and a significant reduction in tuning requirements. Applications running on Violin flash Arrays are unleashed, allowing DBAs and application designers to step back from the excessive demands of performance tuning and concentrate on business requirements.

## Cost-effective Scalability Without Performance Degradation

Under-performing databases lead to lost productivity and directly impact an organization’s bottom line. With Violin’s all-flash Memory Arrays, you get a plug-and-play installation that provides resilient performance, at any scale. Flash memory means that storage access times are many orders of magnitude faster than their disk counterparts, resulting in extreme performance, at a lower \$/GB than disk. Violin’s flash arrays scale with any increase in the number of concurrent users, or consolidation of multiple database or workloads.

## Reduced Total Cost of Ownership

To achieve minimum latency and highest IOPs, hardware is over-provisioned or software is used as a workaround to mitigate but not fully eliminate the issue. These workarounds incur both capital as well as operational expenditures.

Violin flash Memory Arrays, at 3U, allow for the consolidation and scalability of your Oracle databases, all while providing a reduction of:

- 30% - 70% in database licensing costs
- 60% - 90% in power, cooling & storage footprint
- 30% - 80% in processors