

5 Reasons Why Oracle Storage is the Best IT Choice for Oracle Database



Oracle Storage is the only storage co-engineered with Oracle Database to meet its demanding performance, efficiency, security, and management requirements. And, its high-performance unified storage architecture provides a cost-effective and easy to deploy solution that is ready for the cloud whenever you are.

Quite simply, it's the best choice when you are investing in infrastructure to run Oracle Databases

Higher Performance



Oracle Database is at the core of your business, so it needs to run as fast as possible in production environments as well as for development, testing, reporting, and other secondary processing needs.

Only Oracle Storage combines all-flash designs and unique optimizations that are co-engineered with Oracle Database to lower the latency and increase overall database throughput. By tightly integrating Oracle Storage into the Oracle Database stack, you also get more from existing database servers and licenses.



2x more real-world storage performance



Automated optimization of Oracle Database storage



Up to 4x faster application processing

Greater Efficiency

Oracle Databases continue to grow rapidly, and when you couple that growth with the need for more development, testing, and secondary processing copies, the amount of storage you need is astounding.



Oracle Storage maximizes Oracle Database storage efficiency while maintaining Oracle's security best practices. From no-overhead snapshots and clones for development and testing to Oracle only support for Oracle Database Hybrid Columnar Compression, Oracle Storage solutions help reduce the amount of storage you need to buy more than any other storage system.



50% less storage required in typical database environments



90% smaller data warehouses and database archives



4:1 consolidation of legacy storage systems

Simplified Management

With today's shrinking IT staffs, managing Oracle Database storage can be a time-consuming activity – one that today's real-time enterprises literally can't afford.



Oracle Storage is co-engineered with Oracle Database to automate storage management and reduce costs. Oracle Storage uses real-time Oracle Database metadata to dynamically tune itself, and enable DBA-centric interactive storage analytics at the pluggable database level. When using Oracle Storage, IT organizations spend less time "keeping the lights on" and more on business innovation.



10x faster database provisioning for development and testing



65% fewer management steps than competitive systems



ZERO translation needed between DBAs and storage administrators

Cloud Integration

Cloud changes everything, so it's important that your IT infrastructure enable you to integrate on-premises storage with the cloud.

Oracle Storage is designed to meet the demanding needs of cloud environments and their ever-changing mix of applications, databases, and development environments. With powerful on-premises capabilities and complementary Oracle Public Cloud services, you can increase application development agility without added complexity, security risks and hidden costs.



CLOUD Architected to support changing workloads



CLOUD Managed with REST APIs



Cloud Proven with over **1 EXABYTE** in Oracle IT clouds

Better Value



At the end of the day, it's important to ask yourself "Which storage systems reduce today's costs, accelerate the business, and simplify my journey to the cloud?"

Oracle Storage is uniquely able to meet these needs because it is co-engineered with Oracle Database and Oracle Cloud.

It offers greater performance and efficiency so you buy less storage, reduces management costs through automation, and offers a built-in path to the cloud that eliminates hidden integrations and costs.



"The high-end ZFS storage array is the highest performing hybrid storage device that has been analyzed by Wikibon, and in a class of its own when it come to high write-IO environments."

– David Floyer, Wikibon



[Learn More About Oracle ZFS Storage Appliance](#)