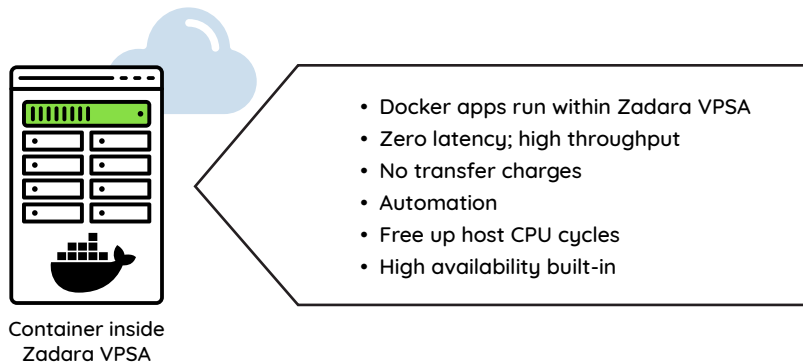


# Container Services

Container Services allow you to put Docker containers inside your storage, for fast processing, task automation and high availability

Containers add versatility, reliability, and performance.

Activate Docker containers inside your Zadara storage for lightning-fast performance, task automation, and application high availability. Zadara’s convergence of Docker container technology into a pair of virtual controllers, makes running processing inside storage possible. The virtual controllers have direct, low-latency access to your SSDs and HDDs, for high-throughput IOPS to the Docker container running within.



## Features & Benefits

### Low-Latency

Direct access to SSDs, HDDs, for high-throughput IOPS

We call the container an app, and like the virtual controller, it has dedicated CPU and memory, which can be increased and decreased on the fly, non-disruptively. Uniquely, Zadara offers Docker high availability, thanks to Zadara’s dual-controller architecture with auto-failover.

### Free VM CPU

Free the virtual machine CPU to perform higher value tasks

### Eliminate technical, operational, and financial risk.

Zadara eliminates the technical, operational, and financial risks associated with enterprise storage, by delivering industry-leading enterprise SAN, NAS and Object storage as a fully-managed service with 24/7 proactive monitoring and support, backed by a 100%-uptime guarantee, and wrapped in a pay-only-for-what-you-use model.

### Dual Controllers

Container has pair of controllers (CPU and RAM)

### Transform your business with zero-risk enterprise storage.

Zadara transforms storage-related costs from a variable mix of equipment and management expenses to a predictable, on-demand, pay-per-use, elastic service that greatly simplifies planning, streamlines budgeting, and improves return on investment (ROI). Find out how zero-risk enterprise storage can help transform your business. Call or email today.

### High Availability

Containers have built-in high availability