StorPool
DISTRIBUTED STORAGE

StorPool Storage

Ultra-fast, Reliable, Easily Scalable Primary Storage Platform

Technical Overview

Table of Contents

Introduction

StorPool Storage™ offers a **radical departure** from dual-controller, shared-disk storage array designs and the software-defined implementations of that architecture. This legacy paradigm has been the default solution for enterprise and service provider IT organizations for decades. It made sense initially, and it still might for simpler deployments with predictable workload requirements. However, it is proving unsuitable for complex multi-petabyte-scale environments that demand extreme input/output (I/O) parallelism, reliability, and low latency at scale. Fleets of standard servers are preferred for large-scale infrastructure deployments like those at cloud service providers. IT organizations require storage solutions that use such servers to consistently meet their performance and data management requirements while providing fault tolerance up to multi-rack failures, a low total cost of ownership (TCO), end-to-end automation, dynamic resource management, tech refresh simplicity, and alignment with application servers. They want to be able to perform deployments, upgrades, and refreshes non-disruptively, on their preferred timeline, and with the hardware of their choice.

StorPool Storage addresses the mismatch between the legacy paradigm and modern IT organizations' needs by working closely with hundreds of forward-thinking companies to develop an ultra-fast **primary storage platform**, which is less expensive to deploy and maintain at scale and outperforms legacy storage solutions. The result is intelligent software that delivers advanced storage features using commodity off-the-shelf servers (COTS).

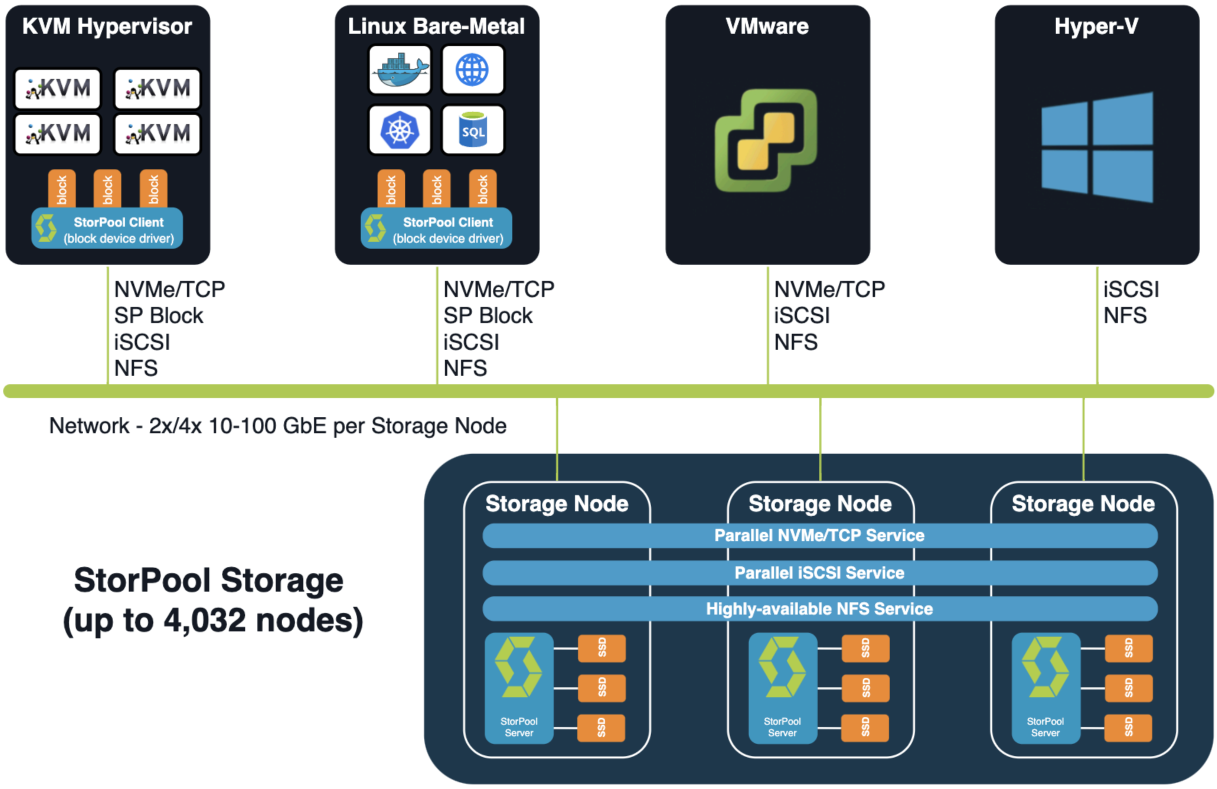

Fig 1. Logical Diagram of StorPool Storage in a Multi-Platform Scenario

StorPool Storage enables deploying and consuming primary storage in **a novel way**. The software's **parallel, multi-node, shared-nothing architecture** enables StorPool storage systems to scale to many petabytes. Individual storage clusters can scale out to 63 nodes. With the software's multi-cluster capabilities, up to 64 sub-clusters can work as a unified large-scale storage system with a global namespace under common management - up to 4,032 nodes and 60+ PB usable capacity per system. Systems grow non-disruptively with no performance bottlenecks or single points of failure.

With StorPool Storage, IT organizations can deploy large-scale clouds rivaling hyperscale cloud service providers. Following the approach of hyperscalers, the software is installed on sets of standard servers running Linux (e.g. RHEL, RHEL clones, Oracle Linux, Ubuntu, Debian), converting them into high-performance, linearly scalable storage systems. StorPool **seamlessly pools the capacity and performance of the storage devices in each set.**

Use Cases

StorPool storage systems are ideal for handling the data of modern workloads demanding extreme reliability and low latency at scale, leading to:

- Significantly reduced business-critical application response times
- Application server and licensing consolidation
- Robust, high-performance virtual machines (VMs)
- Exceptional acceleration of transactional databases
- Ultra-fast vector databases for AI retrieval-augmented generation (RAG)
- Quicker E-commerce response times
- More robust web servers
- More reliable virtual desktops
- Faster, more scalable, and more responsive online gaming platforms

StorPool's versatility enables customers to meet the needs of multiple use cases with a single primary storage system connected to all their compute environments. To help customers scale from single-application systems to multi-data center deployments, StorPool Storage enables fast and efficient changes to workload settings like volume resizing, QoS modification, or on-demand live migration between storage tiers and sub-clusters. This means **IT organizations can easily address changes in user requirements in flight without interruption in services**.

StorPool VolumeCare™ and **StorPool Disaster Recovery Engine** let customers automate the periodic snapshotting and efficient replication of differential changes to lower storage tiers locally or to remote locations for **backup and disaster recovery** purposes.

StorPool Storage easily integrates with a cloud's management layer to automate workflows. To accomplish this, StorPool provides plugins for cloud management platforms like OpenStack, Kubernetes, Oracle Linux Virtualization Manager (OLVM), CloudStack, OpenNebula, Proxmox, OnApp, and custom-built cloud management solutions.

Thanks to the automation built into each plugin, users **seamlessly manage integrated environments from the cloud platforms' user interfaces or purpose-built Web front-ends**.

KVM hypervisors and bare-metal Linux hosts typically use StorPool storage systems with the StorPool Block Protocol. In bare-metal Linux hosts, volumes are attached and used as standard block devices. Volumes appear as block devices under /dev/storpool/* on compute-only and hyperconverged hosts and behave like local storage devices. Since data is widely striped across all storage devices, StorPool Block Protocol clients communicate in parallel with all nodes in a storage cluster.

With VMware, XenServer, and Microsoft Hyper-V environments, volumes are attached as iSCSI targets and then formatted to store virtual machine and container data. StorPool Storage provides an easy-to-use API, CLI, and GUI for managing StorPool storage systems that serve such environments.

Reliability

StorPool Storage protects and manages workload data while the underlying hardware is non-disruptively continuously upgraded and refreshed with new storage devices and nodes as technology advances. Thanks to this approach, IT organizations can provide the storage tiers their workloads need while easily upgrading hardware across generations. Systems can also grow non-disruptively by scaling out with additional storage nodes, or scaling up individual nodes with more or larger storage devices.

Data protection is provided by synchronously writing three (3N) copies of each data block to storage devices. The write algorithm developed by StorPool confirms each write after the needed copies have been written across a storage cluster's fault sets. Fault sets are typically separate servers, and can even be configured as separate racks. Writing redundant copies across rack-level fault sets increases a sub-cluster's availability if a power circuit or top-of-rack switch fails. In this way, **an entire rack of servers may fail, and data is still 100% protected with 100% system uptime**. In addition, in the event of complete datacenter power failure or network downtimes, StorPool storage systems return to operation with 100% data consistency as soon as services are restored.

With StorPool's **Erasure Coding implementation**, the protection level is equivalent to the protection of triple replication. Any encoded data can be recovered in the case of double failures - for example, due to failure of multiple storage devices in two nodes, complete failure of any two nodes caused by component failures, erroneous extraction of storage devices in a second node while one node is down for maintenance, erroneous shutdown of a second node while one node is down for maintenance, or other reasons. At the same time, it is a high-performance implementation that maintains the read and write performance during normal operations at the levels one gets from triple replication - same latency, same max IOPS and max bandwidth. A minor read latency increase is observed during initial conversion and during fault states - 6µs to 30µs per read operation, varying depending on the Erasure Coding scheme and fault state.

StorPool Storage also implements 64-bit end-to-end checksums to provide **end-to-end data integrity** that ensures the data users put in a storage system is the data they get back. In this way, data is protected from both hardware and software mishaps that would otherwise lead to data loss and are common for all systems handling large amounts of data, such as:

- Silent data corruption
- Phantom, partial, or misplaced writes
- Firmware, driver, or kernel issues

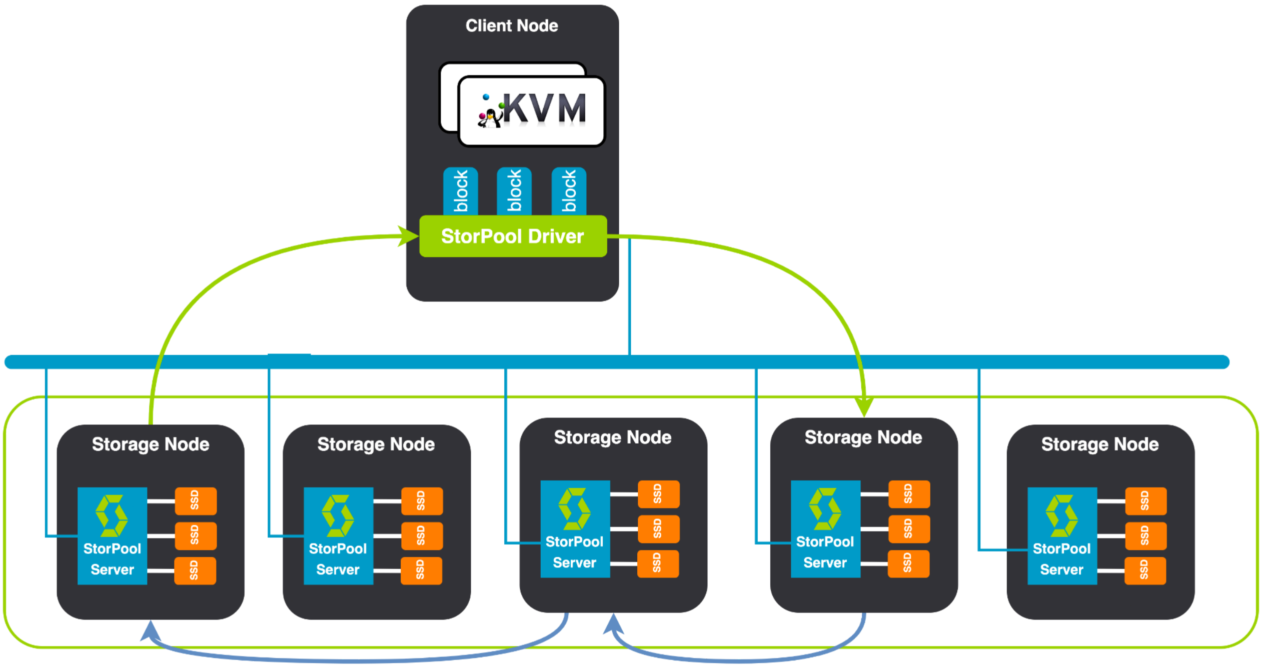

Fig 2. High-Level View of One Synchronous Write Operation from a Single Client

Performance

StorPool storage systems utilizing NVMe SSDs can deliver unmatched speeds:

- Consistent sub-0.1ms latency, even at scale
- Up to 113 million 4KB random read/write 70/30 IOPS per sub-cluster
- Up to 302 GB/s parallel data throughput per sub-cluster

A major and unique benefit of StorPool Storage is that its performance degradation is minimal, which allows the usable capacity utilization to reach as high as 95%. Therefore, customers get at least 20% more real usable space compared to legacy products that tap out at about 75% to 80% utilization.

Thanks to the software's parallel, multi-node, shared-nothing architecture, there are no central I/O processing functions. Therefore, **StorPool storage systems can easily scale linearly** - as drives or servers are added to a storage cluster, IOPS and throughput increase equally **for all data in the system** while latency remains consistently low. Thanks to the wide striping of volumes, StorPool Storage delivers **industry-leading performance during random and sequential write bursts**. It eliminates performance hotspots, removing the additional management overhead experienced with legacy storage products due to the need for continuous workload monitoring, tuning, and rebalancing.

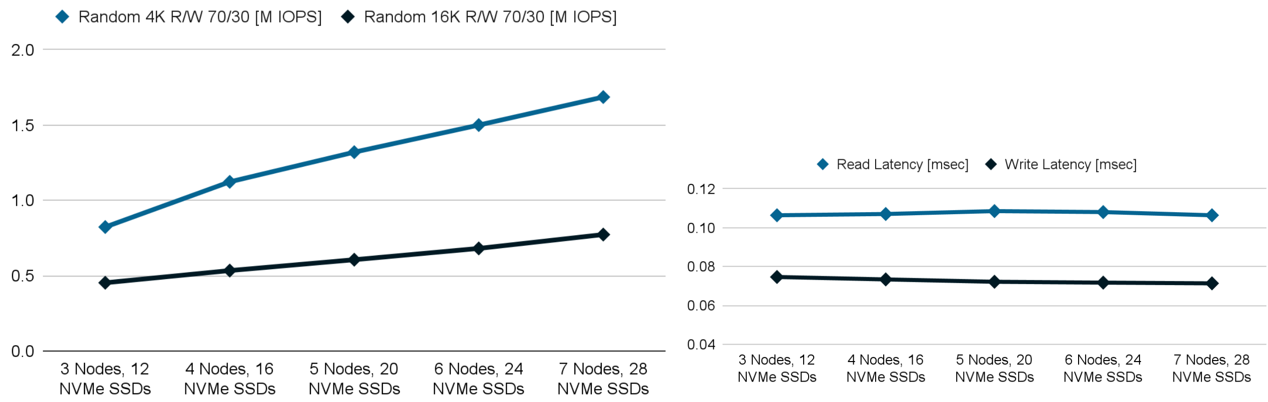

Fig 3. Linear Performance Scalability Example

Cost Efficiency

The cost for SSDs used in StorPool storage systems is up to 90% lower than storage array drives. This is due to several factors. The first is because of unique StorPool technologies enabling efficient use of read-intensive SSDs with 1 drive write per day (1DWPD) warranty, even for workloads like heavily loaded transactional databases. Customers can leverage multiple storage tiers defined with Quality of Service (QoS) settings applied on a per-volume basis, or with separate pools of SATA/SAS/NVMe SSDs and SATA/SAS HDDs. These pools of shared storage resources can **serve up to 6.3 million volumes and snapshots per storage system (100k per sub-cluster)** to end-user workloads in virtual machines, containers, or bare-metal hosts. Each volume is thin-provisioned and widely striped across all the drives in a storage cluster to enable massive I/O parallelization.

Another major cost reduction factor is the use of server drives. Server drives are identical to storage array drives except for the microcode and the cost. The microcode makes sure the customer can only buy drives from the storage array vendor. This empowers the vendor to charge as much as 10 times the price of the server drive. Even after all discounts, server drives are 67% lower cost than storage array drives. There is no difference in reliability, performance, or capabilities. StorPool storage uses server drives.

StorPool Storage additionally extracts extreme performance from standard server components without requiring specialized hardware such as RDMA-enabled NICs, Persistent Memory Modules (PMem), or Data Processing Units (DPUs). All of these factors combine to deliver a very low **initial cost of deployment** and **total cost of ownership (TCO).** In fact, they are so low they're **game-changing**.

Agility

A key StorPool advantage of widely striping volumes across all storage devices is the volume size is not limited by available space on individual drives or nodes. When new storage devices or nodes are added, **data is rebalanced quickly without impacting end-user workloads**. Likewise, rebuild times are limited mainly by the write performance of the drives receiving data. For data redundancy, StorPool supports three copies (3N) and **high-performance erasure coding** technology (N+2). StorPool's erasure coding technology **increases storage efficiency by up to 140%, reducing hardware costs per usable TB by up to 55%**.

StorPool's instant redirect-on-write snapshots and thin clones have no negative performance impact and do not copy underlying data**,** which enables the creation of virtual disk templates and golden images used for:

- cloned provisioning of thousands of workloads
- dev/test/staging use cases
- analytics/reporting use cases

StorPool Storage provides a comprehensive set of tools and services with pay-as-you-grow subscription licensing. The result is a value proposition that traditional storage product vendors struggle to match, especially at scale, backed by several add-on products and services:

- **StorPool Analytics** - a cloud-based analytics tool that collects millions of metrics per second to provide visualizations with deep insights on several levels - per volume, per storage node, per host, and the whole system.
- **StorPool Monitoring** - a cloud-based console that uses various methods to track the availability of end-to-end services and automatically raise alerts about potential issues in customers' fleets of StorPool storage systems.
- **StorPool VolumeCare** - a service that automates the periodic snapshotting and efficient replication of differential changes to lower storage tiers locally or to remote locations for backup and disaster recovery purposes.
- For customers taking advantage of the **StorPool Managed Services**, our in-house team of storage experts designs, deploys, tunes, monitors, manages, and maintains their StorPool storage systems. The StorPool Managed Services team leverages the storage analytics and monitoring suite to proactively open support tickets and deal with potential issues before they impact end-user workloads.

High-Level Architecture

StorPool Storage uses a parallel, multi-node, shared-nothing architecture. It is installed on standard Linux servers (e.g., RHEL, RHEL clones, Oracle Linux, Ubuntu, Debian) and converts them into storage clusters. Depending on the hardware selected for each server, it can be part of primary, secondary, or hybrid storage systems serving various use cases. StorPool Storage performs all functions on all the servers in a cluster on an equal-peer basis. StorPool storage systems can write/read data over four storage protocols:

- **StorPool Block Protocol** - the native StorPool protocol delivers complete control of each StorPool system up to the attach point, end-to-end data integrity, extremely high performance and resource efficiency, and sub-5-second I/O recovery on node failure.
- **NVMe/TCP** - a next-generation block storage protocol that uses standard initiators available in VMware, Linux, and other computing environments to enable high-speed, low-latency access to primary storage systems over Ethernet networks.
- **iSCSI** - legacy block storage protocol that has wide compatibility with virtualized, containerized, and bare-metal environments.
- **NFS** - file storage protocol suitable for shared access of throughput-intensive and moderate-load file workloads.

Per node, StorPool Storage typically uses only 2 to 6 CPU cores and 1 GB RAM per 1 TB of raw storage. This allows deploying StorPool Storage in disaggregated, hyperconverged, and mixed infrastructure environments.

Disaggregated Infrastructure

Customers can buy cheap, resource-efficient servers right-sized as storage nodes to deploy their cloud with a disaggregated architecture. This approach delivers the best value for money in the long term, and makes large-scale clouds easier to manage and maintain. It applies to VMware, OLVM, oVirt, XenServer, Microsoft Hyper-V, and KVM-based environments managed with OpenStack, CloudStack, OpenNebula, Proxmox, OnApp, or custom-built cloud management solutions. The StorPool disaggregated infrastructure enables hyperscale block data storage high-density clusters of up to 63 nodes and up to 64 sub-clusters in a single system.

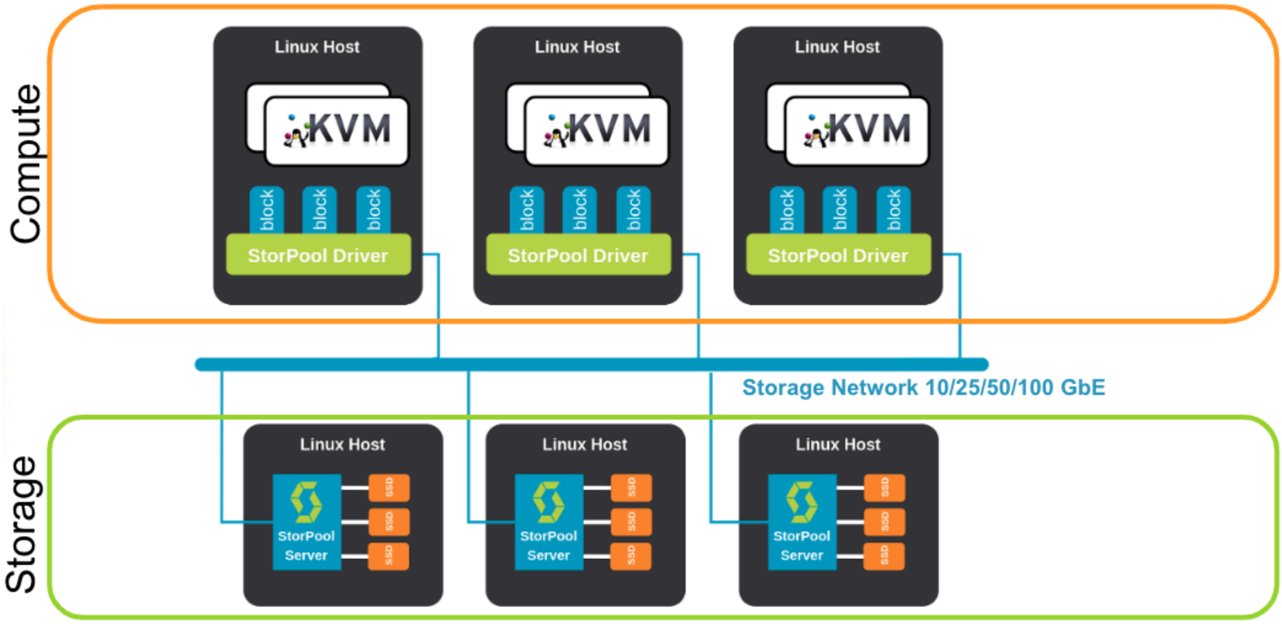

Fig 4. Logical Diagram of Deployment Option 1: Disaggregated Infrastructure

Hyperconverged Infrastructure

Customers can buy mixed-use servers with enough RAM, CPU, and storage capacity to run compute and storage concurrently. This approach delivers the lowest upfront cost and simplifies small- and medium-sized clouds' logical, physical, and network topology. It applies only to environments using the StorPool Block Protocol - bare-metal Linux hosts and KVM hypervisors managed with OpenStack, CloudStack, Oracle Linux Virtualization Manager (OLVM), oVirt, OpenNebula, Proxmox, OnApp, or custom-built cloud management solutions.

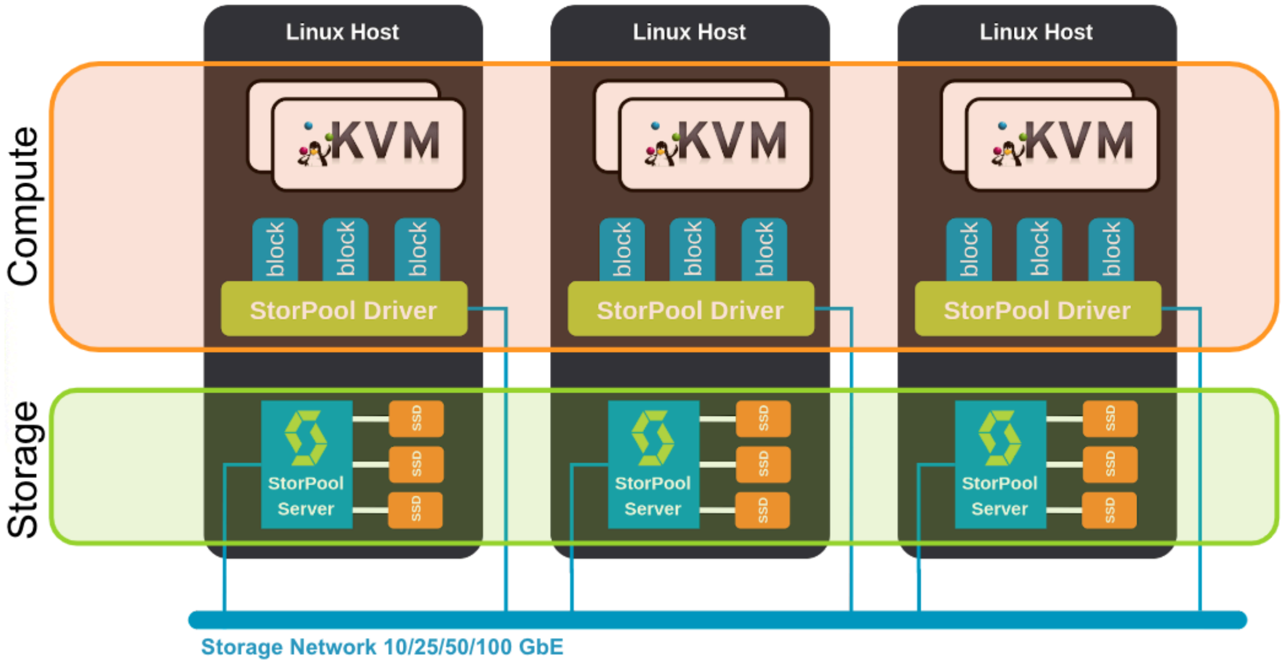

Fig 5. Logical Diagram of Deployment Option 2: Hyperconverged Infrastructure

Mixed Environments

Customers with complex environments can build heterogeneous clouds, where some servers are hyperconverged, some are storage-only, and some are compute-only. StorPool Storage aggregates the capacity and performance of the storage devices in hyperconverged and storage-only nodes to serve both compute-only and hyperconverged hosts, using the storage protocol suitable for each use case.

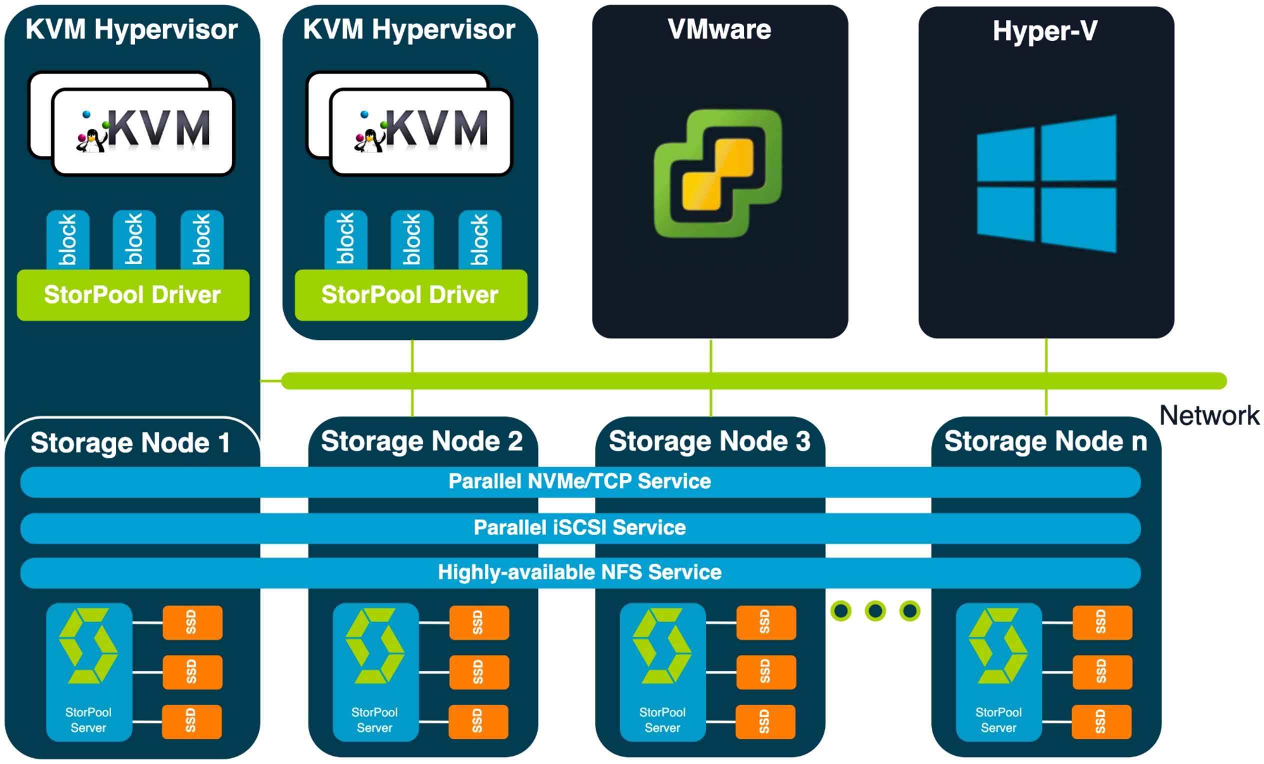

Fig 6. Logical Diagram of Deployment Option 3: Mixed Environments

Each virtual machine, container, or bare-metal host uses one or more volumes created in StorPool Storage. In environments managed with OpenStack, Kubernetes, CloudStack, OpenNebula, Oracle Linux Virtualization Manager (OLVM), oVirt, Proxmox, OnApp, or custom-built cloud management solutions, an integration between the cloud orchestration and storage layers enables automation of all storage actions (e.g., create a virtual machine based on a template, create virtual disk, attach virtual disk, resize virtual disk, create a snapshot, etc.). Actions supported by the respective cloud platform are initiated through its user interfaces and executed automatically on the storage layer.

KVM hypervisors and bare-metal Linux hosts typically use StorPool storage systems with the StorPool Block Protocol. In bare-metal Linux hosts, volumes are attached and used as standard block devices. Volumes appear as block devices under /dev/storpool/* on compute-only and hyperconverged hosts and behave like local storage devices. Since data is widely striped across all storage devices, StorPool Block Protocol clients communicate in parallel with all nodes in a storage cluster.

When used with VMware, XenServer, or Microsoft Hyper-V, StorPool Storage volumes are attached as NVMe/TCP or iSCSI targets and formatted as needed by each platform (e.g., VMFS, NTFS, Windows CSV, CLVM, or OCFS2). The targets are used with the standard initiators available in the respective platform. StorPool NVMe/TCP and iSCSI targets are highly available and benefit from the wide striping, data integrity, and other features inherent in the StorPool Storage backend.

Low-Level Architecture

Over the years, the StorPool Storage team has built a storage solution with best-in-class performance, reliability, agility, and ease of management and maintenance at scale. This section is an introduction to some of the software that makes that possible:

- **StorPool Storage** - converts the servers owned, leased, or rented by customers into high-performance, linearly-scalable primary storage systems that can write/read data over four storage protocols - StorPool Block Protocol, NVMe/TCP, iSCSI, and NFS.
- **StorPool Analytics** - a cloud-based analytics system collecting millions of metrics per second to provide visualizations with deep insights into each StorPool storage system on several levels - per volume, per storage node, per compute host, and whole system.
- **StorPool Monitoring** - a cloud-based monitoring console that uses various methods to track the availability of end-to-end services and automatically raise alerts about potential issues in customers' fleets of StorPool storage systems.
- **StorPool VolumeCare™** - automates the periodic snapshotting and efficient replication of incremental changes to lower storage tiers locally or to remote locations for backup and disaster recovery use cases.

StorPool Storage

StorPool Storage consists of multiple purpose-built components designed, developed, and supported by StorPool's engineers. The components are installed on physical servers running Linux (e.g., RHEL, RHEL clones, Oracle Linux, Ubuntu, Debian). Each physical server is a compute-only host, storage-only node, or hyperconverged node. Storage-only and hyperconverged nodes are part of each storage cluster and collaborate to provide storage services to clients.

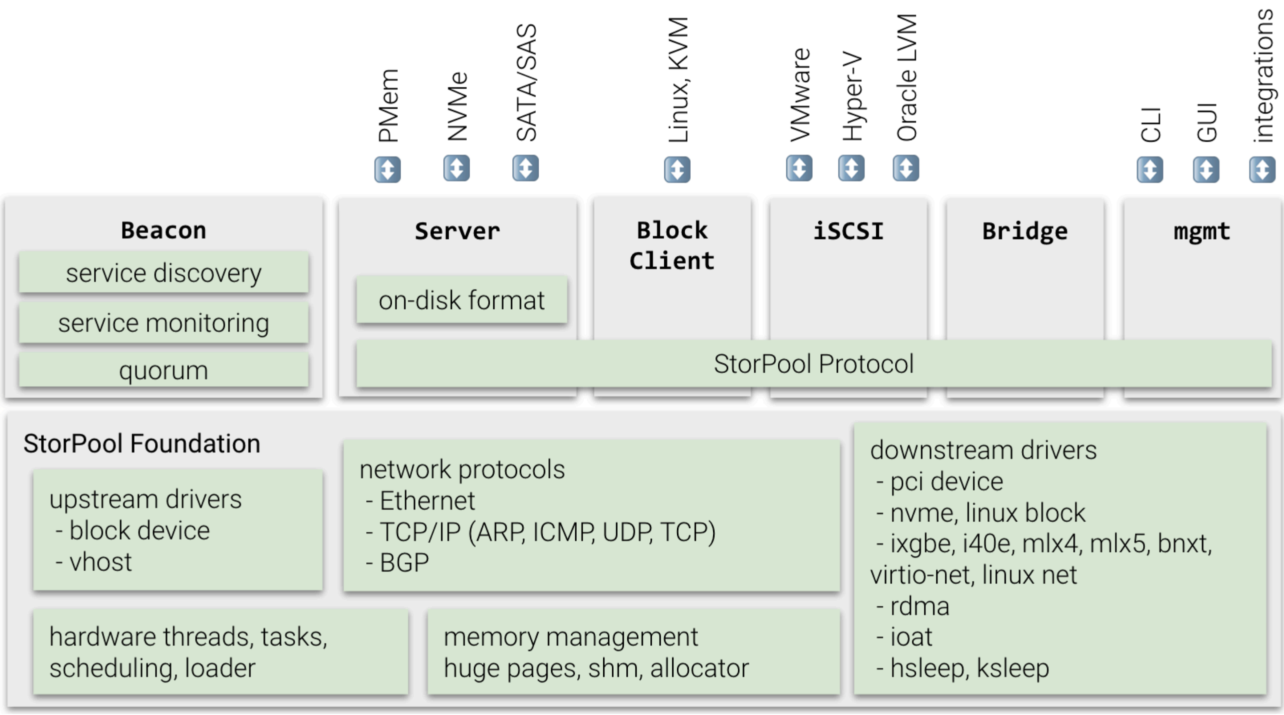

Fig 7. Technical Diagram of Main StorPool Storage Components

Beacon (storpool_beacon)

StorPool Storage has a parallel, multi-node, shared-nothing architecture based on a quorum of nodes that scales with a storage cluster. The storpool_beacon ("beacon") component is the first service installed on all nodes in a storage cluster and informs about the availability of each node.

The StorPool team has built several network protocols that support simple dual-switch networks and routed layer 3 leaf/spine networks with up to 64 storage sub-clusters running in one or more broadcast domains. They allow a network of nodes to communicate reliably and exchange messages between the shared-nothing services running in nodes, so they can be managed as a single integrated entity that provides storage services to clients. The beacon protocol is one of these protocols.

The beacon services communicate over the beacon protocol to ensure that the cluster has quorum, meaning that at least half of the nodes in the cluster plus one are available. StorPool Storage allows the participation of non-storage nodes in the quorum to increase the fault tolerance of small storage clusters with 3 or 4 nodes. For example, in a storage cluster with six beacon services, at least four beacons should be available for the storage cluster to run.

Server (storpool_server)

The storpool_server ("server") component is installed on all nodes, contributing to the storage and performance capacity of a StorPool storage cluster. Storage devices are assigned to the server service(s) in each node. Each node can have up to 12 server services running in parallel to handle write/read operations. Each server service is extremely efficient, handling hundreds of thousands of IOPS with a single dedicated CPU thread.

Since server services work in parallel, performance-optimized StorPool storage systems can handle tens of millions of IOPS and hundreds of GB/s with low latency. Depending on their performance needs, customers can choose to either:

- **Maximize resource efficiency** - have a single server service per node responsible for all the storage devices in that node, or
- **Maximize performance** - have multiple server services per node, each responsible for some of the storage devices in that node.

For example, a node with 24 storage devices would provide better peak performance and performance consistency with 12 server services – each responsible for 2 storage devices – than with a single server service responsible for all 24 storage devices. The number of server services can be changed non-disruptively in production to address changes in requirements.

In addition to handling writes, reads, and data integrity checksums during normal conditions, server services are responsible for re-testing and re-adding storage devices ejected by mistake or by automated checks. Server services also execute background tasks like rebalancing, rebuilding/self-healing, and scrubbing.

Each server service uses the StorPool Storage on-disk format on its assigned storage devices. StorPool's on-disk format is a copy-on-write format with 4k data granularity and is optimized for extremely fast block operations. It is a log-structured format, which means that data and metadata updates necessary for write operations are grouped and performed as a single device operation, ensuring very low write latency without compromising consistency and without the need for specialized hardware like persistent memory modules.

SATA/SAS Drive Operations (storpool_disk)

The storpool_disk kernel module is installed on all nodes that have SATA/SAS SSDs or HDDs. When a server service writes or reads data on SATA/SAS SSDs or HDDs, storage operations to and from each drive pass through the storpool_disk kernel module.

NVMe Drive Operations (storpool_nvmed, storpool_pci, vfio)

The storpool_nvmed component is installed on all nodes that have NVMe SSDs. It is responsible for managing NVMe devices. It implements a user-space poll-mode driver for NVMe SSDs, a device manager with SMART-like functions, and an NVMe queue manager that enables assigning NVMe SSDs to multiple server services simultaneously.

When server services write or read data on NVMe SSDs, the lowest possible latency is delivered by CPU pinning and bypassing the file system and Linux kernel.

The storpool_nvmed service achieves this by unbinding NVMe SSDs from the kernel nvme driver and binding them to storpool_pci or vfio kernel modules, enabling PCI passthrough and direct use of the NVMe devices from user-space drivers in the storpool_nvmed service (for management functions) and server services (for I/O operations). Server services get access to NVMe queues from storpool_nvmed and write/read directly to/from NVMe SSDs.

Block (storpool_block)

In environments using the StorPool Block Protocol, the storpool_block component is installed on all compute hosts, storage, and hyperconverged nodes. The block client service exposes volumes as block devices under /dev/storpool/* in compute hosts and hyperconverged nodes. Thanks to this mechanism, each volume appears to KVM hypervisors and bare-metal Linux hosts like a local storage device. The block client service provides many functionalities that enable the benefits of the native StorPool Block Protocol.

For example, in the write direction, the block client knows which drives each incoming data block should be written to and chooses the server services that should handle each operation. It generates a data integrity checksum for every data block. The user data, metadata, and checksums are sent over the network and stored together, ensuring data integrity. After the server service commits the data to a storage device, it acknowledges the completion of each operation, and the block service confirms the user write operation.

In the read direction, the block client requests data directly from the right server service (avoiding unnecessary network hops), validates that data integrity checksums match the expected values, and completes the user read operation. If the checksums do not match, the block client retries the operations with another copy of the data. If the data in one copy has been corrupted, the storpool_server service schedules the recovery from a copy where the checksums match the expected values.

In addition to the functionalities above, the block client is controlled and monitored through the cluster-level StorPool API for attachments, renames, resizing, and other functions. It also implements purpose-built buffer management, congestion avoidance, retransmission, and other algorithms that ensure the extreme speed and reliability of StorPool storage systems.

iSCSI (storpool_iscsi)

StorPool Storage delivers highly available high-performance iSCSI storage to bare-metal hosts and hypervisors managed with VMware, OLVM, oVirt, XenServer, Microsoft Hyper-V, or OpenStack. Similarly to the block client, the storpool_iscsi service acts as a client that translates write and read operations initiated by compute hosts to the StorPool internal protocols.

In environments using the iSCSI block storage protocol, the storpool_iscsi component is installed on all storage and hyperconverged nodes. This storpool_iscsi service is distinct from other StorPool services by requiring four network interfaces: two for communication within the cluster, and two for providing the iSCSI service to initiators. It features a slimmed-down TCP/IP stack optimized for fast, low-latency networks and supports hardware acceleration. The service ensures high availability and performance by using dedicated physical interfaces for iSCSI traffic, allowing for automatic rebalancing and redirection of targets in case of node failure.

StorPool's implementation of iSCSI is particularly useful in environments where StorPool's native block protocol is not supported (e.g., VMware, Microsoft Hyper-V, etc.) or in complex setups like OpenStack deployed via Juju charms. It provides administrative separation, allowing for a more scalable and flexible approach to client management and supporting more than 64 clients per storage cluster.

The storpool_iscsi service uses floating IP addresses for portal groups to simplify iSCSI initiator configuration. Each portal group is configured with two networks for multipath support, enhancing load balancing and resilience to network failures. Additionally, the storpool_iscsi service supports Layer-3/routed networks, using BGP to announce floating IP addresses for portal groups, facilitating the integration with existing leaf/spine network infrastructure.

Management (storpool_mgmt)

The storpool_mgmt component is installed on multiple nodes in a StorPool storage cluster. It is primarily responsible for listening for incoming HTTP RESTful API calls and authenticating and executing them within the storage cluster. Commands from the command line interface (CLI) and graphical user interface (GUI) are converted to API calls and executed by the active management service. All supported API calls can be reviewed in the StorPool API reference.

Thanks to the management service, each StorPool storage cluster behaves as an integrated entity, and configuration and state are always synchronized and consistent across all nodes. It enables customers to easily:

- Automate storage-related functionalities when working with platforms like OpenStack, Kubernetes, CloudStack, OpenNebula, Proxmox, OnApp, and custom-built cloud management solutions,
- Build self-service user interfaces for end-users to self-provision virtual machines and change workload settings on the fly,
- Generate automatic monthly billing for capacity consumed by end-users,
- Build tools that provide end-users with usage reporting for their environments,
- Automate migrations between virtualization technologies and storage products.

The management service is highly available - if the active management node fails, the service is automatically activated on another node. Without an active management node, the storage cluster will continue to serve write/read operations, but changes like creating volumes and snapshots would be impossible and the system's state is unknowable. Clusters with stalled API requests or an inactive management service trigger a Severity 1 alert in StorPool Monitoring. Such alerts are handled with utmost urgency.

StorPool Analytics

The main goal of the StorPool Analytics system is to allow all StorPool customers to easily and reliably monitor the health and performance of their StorPool storage systems. It is a cloud-based analytics system that consists of multiple purpose-built components designed, developed, and supported by StorPool’s engineers. It also uses several tools for infrastructure monitoring that were selected as the best-of-breed option for each layer of the system (e.g., Grafana, InfluxDB). The ingestion, formatting, and visualization of cluster metrics are performed by dedicated infrastructure in a StorPool-owned datacenter with restricted physical access.

The StorPool Analytics system collects and visualizes a variety of important metrics, like read and write latencies, bandwidth, number of I/O operations per second that are collected with per-second granularity, available on a per-storage-device, per-volume, per-storage-node, per-compute-host, or whole-system basis. The system stores 1-second granularity metrics for 7 days, and, in addition, keeps a per-minute record of all metrics for 2 years to enable longer-term statistical analysis.

The StorPool monitoring agents send all monitoring and performance metrics over encrypted HTTP/TLS connections. Connections are established by the agents on the storage and client nodes with installed StorPool software to a pool of redundant monitoring servers. The TLS connections can be established over the Internet, or using a purpose-built VPN between the storage cluster and StorPool's infrastructure. In both cases, TLS-encrypted communication is used between the agents and the monitoring servers.

The full list of collected metrics can be found in the [StorPool Analytics knowledge base entry](#).

StorPool Monitoring

StorPool Monitoring tracks almost all deployments and provides notifications to customers on issues with their StorPool storage systems. StorPool Monitoring provides alerts across an array of categories, such as cluster status from the StorPool API, per-host alerts from host-level data, and metrics-based alerts based on data from other sources. Alerts can be based on a component, a node, a cluster, or even combinations.

Alerts are defined by severity:

- OK: No alarm.

  Seeing an alert for this state in most cases is a result of a transition from either WARNING or CRITICAL back to OK.
- WARNING: Problems can be expected.
- CRITICAL: Problems imminent or already happening.
- Super-critical (critical with [SC] tag): The cluster is either down or in a state with severe impact on cluster operations (e.g., I/O latency is over expected thresholds). This type of alert generates an automated call to the StorPool operations team.
- UNKNOWN: Fresh information has not been received about the particular service or cluster.

There are two supported channels for sending alarms: email and Slack.

The full list of available alarms can be found in the [StorPool Monitoring knowledge base entry](#).

StorPool VolumeCare

StorPool VolumeCare supports backup use cases by creating and managing atomic, crash-consistent snapshots of volumes backing the virtual disks of virtual machines based on predefined retention policies. VolumeCare leverages the space-efficient, instant differential snapshots of StorPool Storage that don't impact performance and use only actually changed storage capacity. Customers can avoid the complexity of snapshot scheduling and management by using VolumeCare, which manages the backup schedules for volumes and virtual machines.

VolumeCare configuration is stored in the StorPool cluster's key-value store in the form of a section-based configuration file. It can be viewed and edited with the storpool_vcctl tool.

The complete list of available policies can be found in the VolumeCare knowledge base article.

StorPool Disaster Recovery Engine

StorPool Disaster Recovery Engine (DRE) empowers IT service providers, enterprises, and SaaS vendors running their own KVM clouds to perform virtual machine recovery in minutes, ensuring business continuity in the face of disaster scenarios (floods, fires, bombs, mass power outages, etc.). It simplifies the configuration and execution of disaster recovery (DR) services for VMs in KVM-based cloud environments managed with OpenStack, CloudStack, OpenNebula, or Proxmox, providing the following characteristics:

- Significantly reduced recovery point objectives (RPO)—the DRE enables creating VM recovery points as often as every 15 minutes.
- Significantly accelerated recovery time objectives (RTO) in disaster scenarios—the DRE enables quick VM start-up in case the primary site goes offline.

StorPool DRE supports multiple protection models:

- **Active-passive** - virtual machines in a primary site are protected in a recovery site used solely for recovering workloads in a disaster scenario.
- **Bi-directional** - two sites run virtual machines, and both sites send recovery points to the remote site, each acting as both a primary and a recovery site for the other.
- **Many-to-one** - multiple primary sites run workloads that are protected in a single recovery site used solely for recovering workloads in a disaster scenario.
- **Multi-site Mesh** - multiple sites run workloads and protect each other, acting as both primary and recovery sites.

To begin implementing StorPool DRE in your environment, explore the guide for Getting Started with the StorPool DRE.

About Us

StorPool Storage is an ultra-fast, fully managed block storage software platform designed for large-scale clouds. It is the easiest way to convert sets of commodity servers into primary storage systems. The StorPool team has experience working with various clients – Managed Service Providers, Hosting Service Providers, Cloud Service Providers, enterprises, and SaaS vendors. StorPool Storage comes as software, plus a fully managed data storage service that transforms standard hardware into ultra-fast, reliable, and linearly scalable storage systems. Learn more about StorPool Storage and how we accelerate the world by storing data more productively!

Leading IT organizations like Atos, NASA, CERN, Siemens, Deutsche Bourse, European Space Agency, Samsung, and many others have data running on StorPool.

With a global list of customers, over 20 major releases, and solid growth, StorPool Storage is a reliable yet flexible partner for any public or private cloud builder.

Contact Us

To learn more or schedule a free test, contact us at:

StorPool Storage
info@storpool.com
+1 415 539 0356
www.storpool.com