White Paper

Hitachi Content Software for File

Storage without compromise.

Hitachi Vantara

Hitachi Content Software for File solves common storage challenges by eliminating the chokepoints that impact application performance.

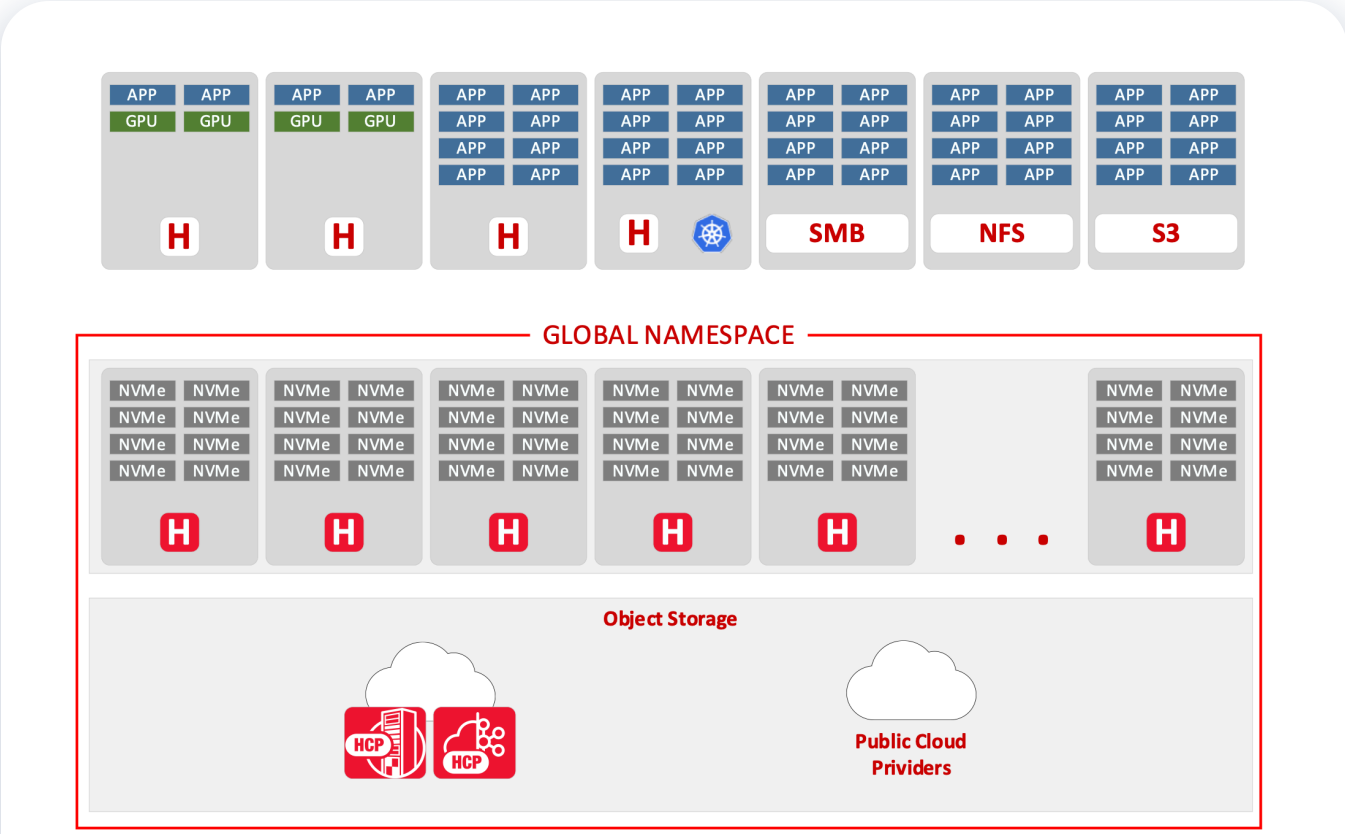
It is well-suited for demanding environments requiring shareable storage with low latency, high performance, and cloud scalability.

Example use cases include:

- Artificial Intelligence (AI) and Machine Learning (ML), including AIOps and MLOps
- Life sciences, including genomics, Cryo-EM, pharmacometrics (NONMEM, PsN)
- Financial trading, including back testing, time-series analysis, and risk management
- Engineering DevOps
- Electronic Design and Automation (EDA)
- Media rendering and visual effects (VFX)
- High-Performance Computing (HPC)
- GPU pipeline acceleration

By leveraging existing technologies in new ways and augmenting them with engineering innovations, Hitachi Content Software for File, powered by WEKA's software delivers a more powerful and simpler solution that would have traditionally required several disparate storage systems. The resulting software solution delivers high performance for all workloads (big and small files, reads and writes, random, sequential, and metadata-heavy). Furthermore, because it is designed to run on commodity server infrastructure, it does not rely on specialized hardware.

Hitachi Content Software for File is a fully distributed parallel file system that is written to deliver the highest-performance file services by leveraging NVMe Flash. The software also includes integrated tiering that seamlessly expands the namespace to and from hard disk drive (HDD) object storage without needing special data migration software or complex scripts; all data resides in a single namespace for easy access and management. The intuitive graphical user interface allows a single administrator to manage exabytes of data quickly and easily without any specialized storage training.



Hitachi Content Software for File's unique architecture is different from legacy storage systems, appliances, and hypervisor-based software-defined storage solutions because it not only overcomes traditional storage scaling and file sharing limitations but also allows parallel file access via POSIX, NFS, SMB, S3, and GPUDirect Storage. It provides a rich enterprise feature set, including local and remote snapshots to an S3 provider, automated tiering, dynamic cluster rebalancing, backup, encryption, authentication, key management, user groups, quotas, etc.

Hitachi Content Software for File Benefits

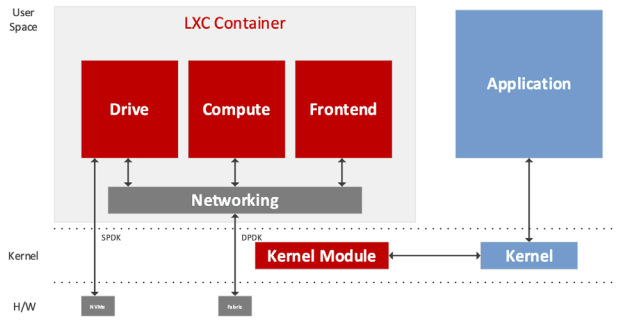
- Highest performance across all I/O profiles – ideal for mixed small and large file workloads
- Scalable capacity – start as small as 100TB and scale to hundreds of petabytes in a single namespace
- Strong security – keep data safe from threats or rogue actors with both software and hardware encryption
- Hybrid Cloud – burst to all the major cloud providers for compute agility or run natively in the cloud
- Backup – push backups straight to HCP, HCP Cloud Scale, or public cloud for long-term retention
- Best economics – combine flash and disk for best cost at scale

Hitachi Content Software for File Architecture

Legacy parallel file systems overlay file management software on top of block storage, creating a layered architecture that can impact performance. Hitachi Content Software for File is a distributed, parallel file system that eliminates the traditional block-volume layer managing underlying storage resources. This integrated architecture does not suffer the limitations of other shared storage solutions and effectively delivers both scalability and performance.

The diagram below provides an overview of the software architecture from the application layer all the way to the physical persistent media layer. The file system core components, including the WekaFS unified namespace and other functions such as virtual metadata servers (MDSs), execute in user space in a Linux container (LXC), eliminating time-sharing and other kernel-specific dependencies. The notable exception is the Virtual File System (VFS) kernel driver, which provides the POSIX file system interface to applications. Using the kernel driver provides significantly higher performance than what can be achieved using a FUSE user-space driver, and it allows applications that require full POSIX compatibility to run on a shared storage system.

Software Storage Architecture



Hitachi Content Software for File supports all major Linux distributions. It leverages virtualization and low-level Linux container techniques to run its own Real-Time Operating System (RTOS) in user space alongside the original Linux kernel. Hitachi Content Software for File manages its assigned resources (CPU cores, memory regions, network interface cards, and SSDs) to provide process scheduling and memory management and to control the I/O and networking stacks. By not relying on the Linux kernel, Hitachi Content Software for File minimizes context switching, resulting in a shorter I/O path and predictable low latencies. It also allows upgrading the HSF backend storage services independently of Linux OS and WEKA client (frontend) upgrades.

Hitachi Content Software for File functionality running in its RTOS is comprised of the following software components:

- File Services (frontend) – manages multiprotocol connectivity
- File System Compute and Clustering (backend) – manages data distribution, data protection, and file system metadata services
- SSD Drive Agent – transforms the SSD into an efficient networked device
- Management Process – manages events, CLI, statistics, and call-home capability
- Object Connector – read and write to the object store

Hitachi Content Software for File core software in the RTOS runs inside LXC containers, which has the benefit of improved isolation from other server processes. When deployed, Hitachi Content Software for File software is containerized as microservices. Multiple containers for SMB, NFS, S3, and core WekaFS may exist per host. By spanning multiple LXC containers, Hitachi Content Software for File enables even greater parallelism and the ability to use more CPU cores and RAM than a single LXC container.

A VFS driver enables WekaFS to support full POSIX semantics and leverages lockless queues for I/O to achieve the best performance while enhancing interoperability. The POSIX file system has the same runtime semantics as a local Linux file system (e.g., Ext4, XFS, and others), enabling applications that previously could not run on NFS shared storage because of POSIX locking requirements, MMAP files, performance limitations, or other reasons. These applications will enjoy massively improved performance compared to the local file system.

Bypassing the kernel means that WEKA's software stack is not only faster with lower latency but also portable across different bare-metal, VM, containerized, and cloud instance environments.

Resource consumption is often problematic with traditional software-based storage designs because these solutions take over the entire server or share common resources with applications. This extra software overhead introduces latency and steals precious CPU cycles. By comparison, Hitachi Content Software for File only uses the resources that are allocated to it inside its LXC containers, which means it can consume as little as one server core and a small amount of RAM in a shared environment (converged architecture-application and storage software sharing the same server) or as much as all the resources of the server (a dedicated appliance). The same software stack is utilized in either case.

File System Design

From the outset, Hitachi Content Software for File was designed to solve many of the problems inherent with legacy scale-out NAS solutions. One of the key design considerations was to build a software platform that could address the requirements of different user groups within an organization at scale or a multi-tenant environment. The most popular scale-out NAS file systems support a construct of a single file system and a single namespace, utilizing directories and quota systems to allocate resources and manage permissions. While this solution worked at a smaller scale, it has made management complex when the number of users and/or directories scale. Complete isolation of user groups requires the creation of new file systems and namespaces, which then create islands of physical storage to manage. Additionally, directory scaling is a problem and typically requires creating multiple directories to maintain performance, further exacerbating the complexity.

As such, Hitachi Content Software for File differs from other scale-out NAS solutions in that it embraces the concept of many file systems within the global namespace that share the same physical resources. Each file system has its own "persona" and can be configured to provide its own snapshot policies, tiering to object stores, organizations, role-based access control (RBAC), quotas, and much more. The file system is a logical construct, and unlike other solutions, the file system capacity can be changed on the fly. Mounted clients can immediately observe the change in file system size without pausing I/O. As already mentioned, each file system has a choice to tier to an object store, and if it is a tiered file system, the ratio of hot (NVMe) tier and object (HDD) tier can also be changed on the fly. A file system can be split into multiple organizations managed by their administrator.

A single file system can support billions of directories and trillions of files, delivering a scalability model more akin to object stores than NAS systems. Directories scale with no loss in performance. Hitachi Content Software for File currently supports up to 1024 file systems and up to 24,000 snapshots in a single cluster.

Hitachi Content Software for File limits:

- Up to 6.4 trillion files or directories
- Up to 14 Exabytes of managed capacity in the global namespace
- Up to 6.4 billion files in a directory
- Up to 4 petabytes for a single file

Supported Protocols

Clients with the appropriate credentials and privileges can create, modify, and read data using the following protocols:

- POSIX
- NVIDIA® GPUDirect® Storage (GDS)
- NFS (Network File System) v3 and v4.1
- SMB (Server Message Block) v2 and v3
- S3 (Simple Storage Service)

Data written to the file system from one protocol can be read via another, so it is fully shareable among applications.

Hitachi Content Software for File Storage Servers

Hitachi Content Software for File storage servers is created by installing Hitachi Content Software for File on any standard AMD EPYC or Intel Xeon Scalable Processor-based hardware with the appropriate memory, CPU processor, networking, and NVMe solid-state drives. A minimum configuration of 8 storage servers is required to create a cluster that can survive a two-server failure. To create an appliance-like experience, Hitachi Content Software for File has worked with hardware vendors to create a single part number that can be used to order a complete storage system, including a software license, operating system license, and hardware support.

Integrated Flash and Disk Layers for Hybrid Storage

The Hitachi Content Software for File storage design consists of two separate layers: an NVMe SSD-based flash layer that provides high-performance file services to the applications and an optional S3-compatible object storage layer that manages the long-term, such as HCP or HCP Cloud Scale. The two layers can be physically separate but logically serve as one extended namespace for the applications. Hitachi Content Software for File expands the namespace from the NVMe flash layer to the object store, presenting a single global namespace that scales to exabytes. As we will see later, Hitachi Content Software for File leverages components of the object store capability to enable cloud bursting, backup to the cloud, DR to another Hitachi Content Software for File cluster, or file system cloning.

Networking

Hitachi Content Software for File supports the following types of networking technologies:

- InfiniBand (IB) HDR and EDR
- Ethernet – 40Gbit minimum, 100Gbit and above recommended

The available networking infrastructure dictates the choice between the two, as Hitachi Content Software for File delivers comparable performance on either one. For networking, the Hitachi Content Software for File system does not use standard kernel-based TCP/IP services but a proprietary networking stack based on the following:

- DPDK maps the network device in the user space and uses the network device without any context switches and without copying data between kernels. This bypassing of the kernel stack eliminates the consumption of kernel resources for networking operations and can be scaled to run on multiple hosts. It applies to both backend and client hosts and enables the Hitachi Content Software for File system to fully saturate up to multiple 200Gbit Ethernet or InfiniBand links.
- Implementing a proprietary WekaFS protocol over UDP (i.e. the underlying network) may involve routing between subnets or any other networking infrastructure that supports UDP. Clients can be on different subnets as long as they are routable enough to reach the storage nodes.

DPDK delivers operations with high throughput and extremely low latency. Low latency is achieved by bypassing the kernel and sending and receiving packages directly from the NIC. High throughput is achieved because multiple cores in the same host can work in parallel, eliminating any common bottleneck.

For legacy systems that lack support for SR-IOV (Single Root I/O Virtualization) and DPDK, Hitachi Content Software for File defaults to the in-kernel processing and UDP as the transport protocol. This mode of operation is commonly referred to as the 'UDP mode' and is typically used with older hardware such as the Mellanox CX3 family of NICs and virtualized hosts.

In addition to being compatible with older platforms, the UDP mode does not dedicate CPU resources but will yield CPU resources to other applications. This can be useful when the extra CPU cores are needed for other purposes.

For RDMA-enabled environments, common in GPU-accelerated computing, Hitachi Content Software for File supports RDMA for InfiniBand and Ethernet to supply high performance without dedicating cores to the WEKA front-end processes.

Application clients connect to the Hitachi Content Software for File storage cluster via Ethernet or InfiniBand connections. The Hitachi Content Software for File software supports 10GbE, 25GbE, 40GbE, 50GbE, 100GbE, and 200GbE Ethernet networks, along with EDR, 200Gb HDR, and 400Gb NDR InfiniBand networks. For the best performance outlined in this document, Hitachi Content Software for File recommends using at least 100Gbit network links.

Many enterprise environments have a mixed network topology composed of Infiniband and Ethernet to support high-performance computing application clients and more traditional enterprise application clients. Hitachi Content Software for File allows InfiniBand and Ethernet clients to access the same cluster in these mixed networking environments, allowing all applications to leverage a high-performance file system.

Hitachi Content Software for File supports VMXNET3 networking from VMware. When the Hitachi Content Software for File client is deployed inside a guest OS in a VMware ESX hypervisor, this ensures that when a vMotion from one ESX server to another occurs, the Hitachi Content Software for File client will continue to remain connected to the Hitachi Content Software for File storage. A list of supported NICs that work with Hitachi Content Software for File is available on each version's release notes:

https://docs.hitachivantara.com/search/all?query=Hitachi Content Software for File&filters=PublicationType~%2522Release+Notes%2522&content-lang=en-US

Network High Availability (HA)

Hitachi Content Software for File supports high availability (HA) networking to ensure continued operation should a network interface card (NIC) or network switch fail. HA performs failover and failback for reliability and load balancing on both interfaces and is operational for Ethernet and InfiniBand. For HA support, the Hitachi Content Software for File system must be configured with no single component representing a single point of failure. Multiple switches are required, and hosts must have a connection to each switch. HA for clients is achieved by implementing two network interfaces on the same client.

Hitachi Content Software for File also supports the Link Aggregation Control Protocol (LACP), starting with Hitachi Content Software for File 4.2 and, later on, the computing clients on Ethernet (modes 1 and 4) for a single dual-ported NIC. Additionally, Hitachi Content Software for File supports failover of Infiniband to Ethernet within the storage cluster to maintain high availability in case the Infiniband network fails. This failover does not apply to clients that may be on one type of network or the other.

The Hitachi Content Software for File file system can easily saturate the bandwidth of a single network interface card (NIC). For higher throughput, it is recommended to leverage multiple NICs. A non-LACP approach sets a redundancy that enables the Hitachi Content Software for File software to use two interfaces for HA and bandwidth, respectively.

When working with HA networking, it is preferred to have the system send data between hosts through the same switch rather than using the switch interconnect (ISL). The Hitachi Content Software for File system achieves this through network port labeling, ensuring ease of use. This can reduce the overall traffic in the network.

Note: Unlike RoCE implementations that require Priority-based Flow Control (PFC) to be configured in the switch fabric, WEKA does not require a lossless network setting to support its NVMe-over-fabrics implementation and can even deliver this level of low latency performance in public cloud networks.

Protocols

Hitachi Content Software for File supports full multiprotocol and data-sharing capability across various protocols, allowing diverse application types and users to share a single data pool. Unlike other parallel file systems, Hitachi Content Software for File does not require additional management server infrastructure to deliver this capability. The following list includes all currently supported protocols:

- POSIX for local file system support
- NVIDIA GPUDirect Storage (GDS)
- NFS for Linux
- SMB for Windows
- S3 for Object access

POSIX

The Hitachi Content Software for File client is a standard, POSIX-compliant file system driver installed on application servers that enables file access to Hitachi Content Software for File file systems. Like any other file system driver, the Hitachi Content Software for File client intercepts and executes all file system operations. This enables Hitachi Content Software for File to provide applications with local file system semantics and performance while providing a centrally managed, sharable, and resilient storage platform. Hitachi Content Software for File provides advanced capabilities such as byte-range locks. It is tightly integrated with the Linux operating system page cache, which is covered later in the caching section.

The WEKA POSIX client provides the highest IOPS, bandwidth, and metadata performance at the lowest latency.

NVIDIA GPUDirect Storage

GPUDirect Storage is a protocol developed by NVIDIA to improve bandwidth and reduce latency between the server NIC and GPU memory, leveraging RDMA.

NFS

The NFS protocol allows remote systems to access the Hitachi Content Software for File file system from a Linux client without the Hitachi Content Software for File client. While this implementation will not deliver the performance of the POSIX client, it provides a simple way to deploy and share data from the storage cluster. Hitachi Content Software for File currently supports NFS v3 and NFS v4.1

SMB

The SMB protocol allows remote systems to connect to shared file services from a Windows or macOS client. The protocol provides a scalable, resilient, and distributed implementation of SMB, supporting a broad range of SMB capabilities, including:

- User authentication via Active Directory (Native and mixed mode)
- POSIX mapping (uid, gid, rid)
- UNIX extension
- SHA 256 signing
- Expanded identifier space
- Dynamic crediting
- Durable opens for handling disconnects
- Symbolic link support
- Trusted domains
- Encryption
- Guest access
- Hidden shares
- SMB ACLs
- Conversion from Windows to POSIX ACLs

S3

Many Web-based applications now support the S3 protocol. Applications like real-time analytics on IoT data can benefit from high-performance S3 access. Hitachi Content Software for File has implemented an S3 front-end support on the file system to accelerate S3 storage I/O. Hitachi Content Software for File delivers huge performance gains for small file I/O accessed via S3. The S3 API on Hitachi Content Software for File supports the following calls:

- Buckets (HEAD/GET/PUT/DEL)
- Bucket Lifecycle (GET/PUT/DEL)
- Bucket Policy (GET/PUT/DEL)
- Bucket Tagging (GET/PUT/DEL)
- Object (GET/PUT/DEL)
- Object Tagging (GET/PUT/DEL)
- Object Multi-parts (POST Create/Complete, GET/PUT/DEL, GET Parts)

In addition, implementing Hitachi Content Software for File S3 supports multiprotocol access. TLS has full S3 audit logs and bucket-level features such as policies, quotas-per-bucket, and Expiry rules for information lifecycle management.

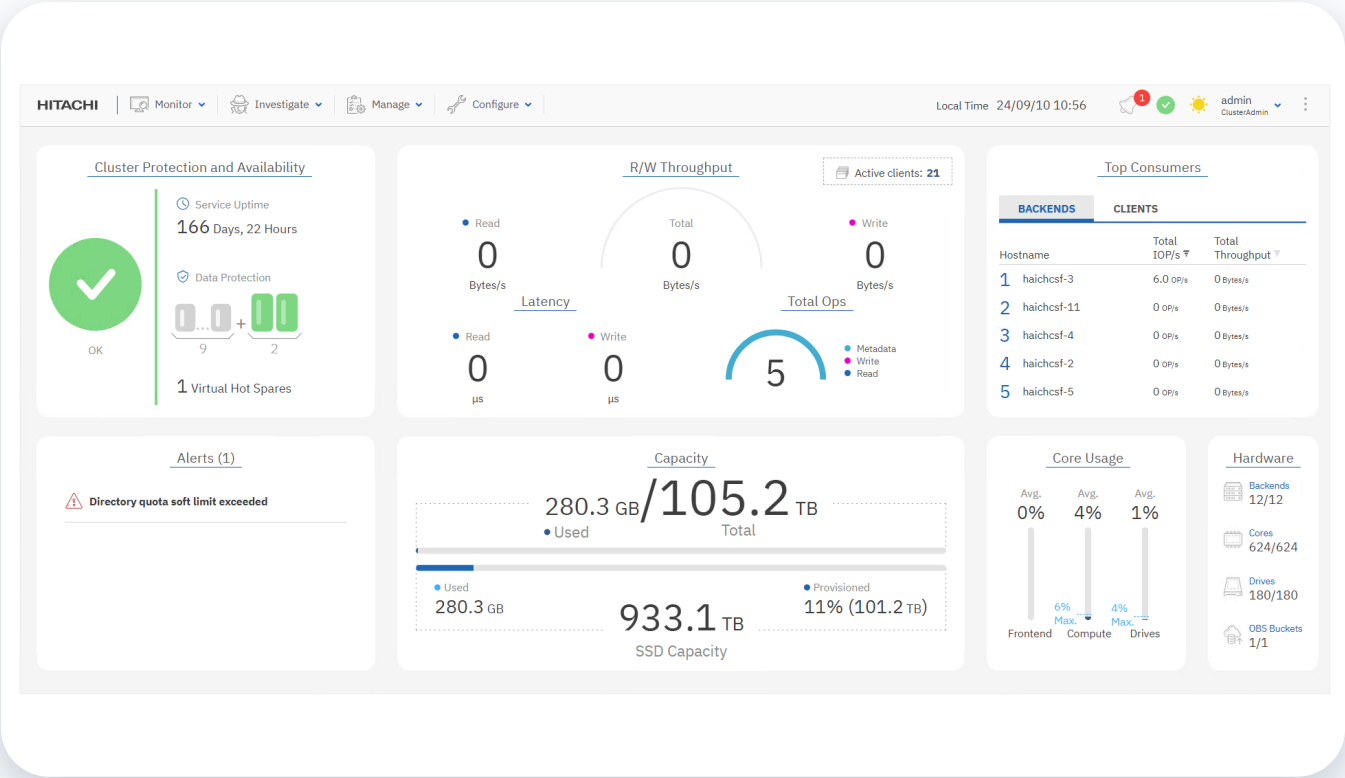
Management GUI

Hitachi Content Software for File provides three quick and easy ways to manage the cluster:

- Graphical User Interface (GUI)
- Command Line Interface (CLI)
- REST API

Reporting, visualization, and overall system management functions are accessible using the REST API, CLI, or the intuitive GUI-driven management console.

Point-and-click simplicity allows users to rapidly provision new storage, create and expand file systems within a global namespace, establish tiering policy, data protection, encryption, authentication, permissions, NFS, SMB, and S3 configuration, read-only or read-write snapshots, snapshot-to-objects, and quality of service policies, as well as monitor overall system health. Detailed event logging allows users to view system events and status over time or drill down into event details with point-in-time precision via the time-series graphing function.



Command Line Interface (CLI)

All Hitachi Content Software for File system functions and services can be executed via a CLI command. Most Hitachi Content Software for File system commands are system-wide and deliver the same result across all cluster nodes. Some commands, such as IP address management, are executed on specific nodes.

REST API

All Hitachi Content Software for File cluster system functions and services can be executed via a web service API, which adheres to the RESTful API architecture. Like the CLI, most RESTful commands are system-wide and deliver the same results on all cluster nodes. Some commands are executed on specific nodes. The API is presented via a Swagger interface for ease of use, as well as examples of API code in multiple programming languages.

Adaptive Caching

Applications, particularly those with small files and lots of metadata calls, benefit significantly from local caches. The data is available with very low latency, reducing the load on the shared network and the backend storage. The Hitachi Content Software for File file system provides a unique advanced caching capability, called adaptive caching, that allows users to fully leverage the performance advantages of Linux data caching (page cache) and metadata caching (dentry cache) while ensuring full coherency across the shared storage cluster. NFS v3 does not support coherency, so utilizing Linux caching can lead to data inconsistency in the read cache and potential data corruption in the write cache. Hitachi Content Software for File supports leveraging Linux page cache — typically reserved for direct attached storage (DAS) or file services run over block storage — on a shared networked file system while maintaining complete data consistency.

The intelligent adaptive caching feature will proactively inform any client that was an exclusive user of a file (and hence running in local cache mode) that another client now has access to the data set. Once this flag is set, the client can continue running in local cache mode until another client modifies the file. The file system will then invalidate the local cache, ensuring that both clients only access the most recent iteration of the data. This maintains the highest performance from the local cache when appropriate and always ensures full coherency on data. This functionality does not require specific mount options to leverage the local page cache as the file system dynamically manages caching, making the provisioning of the Hitachi Content Software for File environment very simple to manage, eliminating the danger of an administrative error causing data corruption.

Hitachi Content Software for File provides the same capability for metadata caching, also known as Linux dentry cache. A client can leverage local metadata cache for a directory, reducing latency significantly. However, once another client has access to the same directory, Hitachi Content Software for File will ensure that any directory changes from one client will invalidate the cached metadata for all other clients accessing that directory. The caching capability also includes extended attributes and access control lists (ACLs).

Caching is typically disabled by default and requires an administrator to change the mount option. That is because write coherency typically depends on some form of battery backup protection on the client to ensure data consistency on a committed write. Hitachi Content Software for File's caching implementation will work out-of-the-box without any administrator intervention as Hitachi Content Software for File does not depend on battery protection to protect acknowledged writes. This feature is ideal for use cases such as "Untar," which will run significantly faster as a local process than across a shared file system.

Global Namespace and Expansion

Hitachi Content Software for File manages all data within the system as part of a global namespace. It supports two persistent storage tiers in a single hybrid architecture — NVMe SSD for active data and HDD-based object storage for a data lake. Expanding the namespace to an object store is an optional addition to the global namespace and can be configured with a few clicks from the management console.

A file resides on flash while active or until it is tiered off to object storage based on preset or user-defined policies. When a file is tiered to the object store, the original file is kept on the flash layer until new data requires the physical space and hence acts as a cached file until overwritten. When file data is demoted to the object store tier, the metadata remains locally on the flash tier, so all files are available to applications in the location they were written to, irrespective of tiering placement, even if the object store bucket was in the public cloud. As NVMe flash system capacity is consumed and usage reaches a high watermark, data is dynamically pushed to the object tier, so you never have to worry about running out of capacity on the flash tier. This is particularly useful for write-intensive applications, as no administrator intervention is required. The flash and object tiers can scale independently depending on the required usage capacities.

The global namespace can be subdivided into 1024 file systems, and file system capacity can be expanded at any time on the fly without the need to unmount and remount the file system. By segmenting the namespace, storage capacity can be allocated to individual users, projects, customers, or other parameters yet be easily and centrally managed. Data within a file system is fully isolated from other file systems to prevent security access issues.

Thin Provisioning

Hitachi Content Software for File allows thin provisioning of file systems within the global namespace. When additional nodes are added to the cluster, any available capacity can be pooled and accessed as a thin provisioned resource. This feature is key in allowing automatic capacity expansion when nodes or drives are added and managing space if nodes or drives are removed from the cluster. After removing nodes or drives, the available capacity must be enough to support the data stored in the flash tier for all the file systems. This feature also enables seamless integration with EC2 auto-scaling groups in AWS and auto-scaling capabilities in other hyperscalar clouds such as GCP, OCI, and Azure.

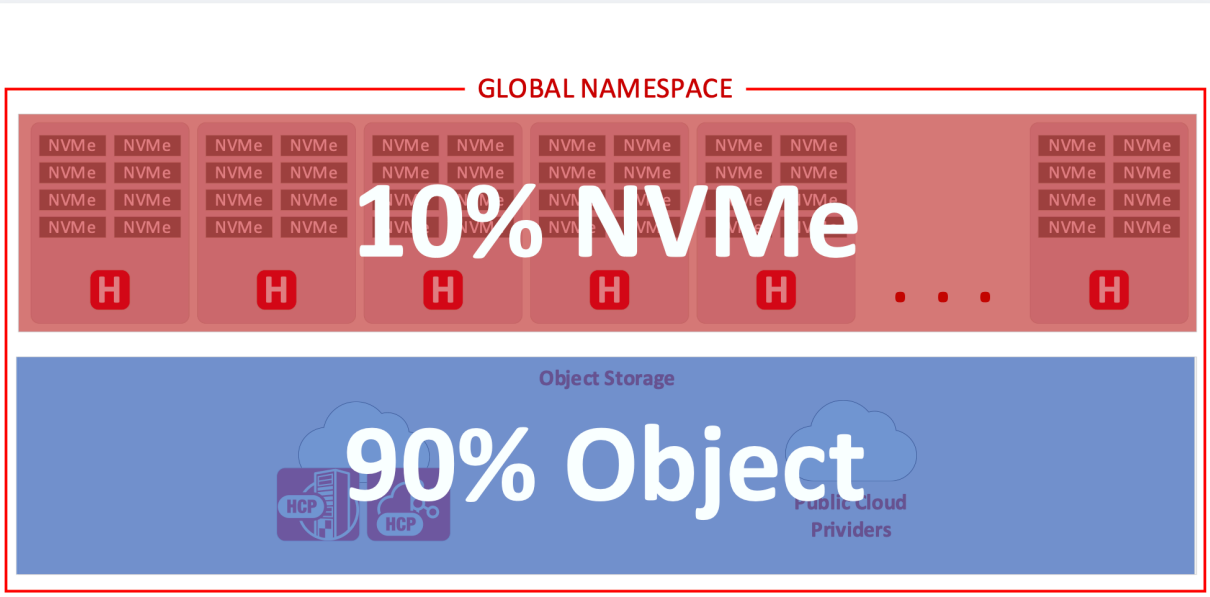
Non-Disruptive Upgrades

Hitachi Content Software for File has the ability to be upgraded without impacting clients. Because it uses containers and sets of processes inside these containers as its OS, Hitachi Content Software for File can upgrade containers in a rolling process without causing a remount of the clients. When combined with the resiliency of its data protection schema, this capability allows for Hitachi Content Software for File to be upgraded on the fly with only a minimum of I/O pauses during the process. Clients need not be unmounted; they can be remounted during the upgrade process.

Integrated Tiered Data Management

Hitachi Content Software for File has a built-in, policy-based automated data management feature that transparently moves data across storage types according to the data temperature. The software supports moving data from the NVMe flash storage tier to HCP and HCP CloudScale or cloud-based object storage. Data movement is set at the per-file system level and is an optional extension of the NVMe flash tier.

For example, users may want to keep specific file systems exclusively on NVMe SSD to ensure the highest performance. In contrast, other file systems implement data movement to object storage for the best cost economics. Metadata is always stored on the flash tier, and a read-only or read-write snapshot of the entire file system, including its data structures, can be stored on the object storage tier to protect against a failure on the flash tier. The clients always see the files in a given file system in the location they were written, regardless of their tiering status. Thus, no change in application is needed to leverage cost-optimized HCP and HCP CloudScale object solutions.



Snapshots and Clones

Hitachi Content Software for File supports user-definable snapshots for routine data protection, including backup. For example, these snapshots can back up files locally on the flash tier and make copies to cloud storage tiers for backup or disaster recovery. Snapshots can be saved to lower-cost cold storage, such as HCP and HCP Cloud Scale object storage. In addition to point-in-time snapshots, Hitachi Content Software for File can create full clones of the file system (read-only snapshots that can be converted into writable snapshots) with pointers back to the originating data. Snapshots and clones occur instantaneously and are incremental after the first instance, dramatically reducing the time and storage required for protection.

Furthermore, system performance is unaffected by the snapshot process or when writing to a clone. Snapshots can be created from the GUI, CLI, or REST API call. Hitachi Content Software for File supports:

- Read-only snapshots
- Read/write snapshots
- Delete the primary snapshot, keeping all other versions
- Delete any snapshot, keeping previous and later versions
- Convert read-only to read/write snapshots
- Snap-to-object (see next section)

Snapshots are exposed to clients via a /.snapshot directory. If tiering is enabled, the snapshot data will be moved to the object tier based on the same policies as the active file system.

Snap-to-Object

Once tiering is enabled in a file system, Hitachi Content Software for File supports a unique feature called snap-to-object. This feature enables committing all the data of a specific snapshot, including metadata, to an object store. Unlike data lifecycle management processes using tiering, this feature involves copying all the snapshot contents, including all files and metadata, to an object store. After the first snap-to-object has been completed, subsequent snapshots are stored incrementally, so backup time is limited to just the changes and is very fast. Hitachi Content Software for File also supports sending snapshots to a second object store using the Remote Backup feature. This leverages the incremental nature of snapshots by only sending the changes across the wire to the destination object store. The object store then only needs to store the incremental capacity of the snapshots at any given time instead of the complete capacity of each uploaded snapshot.

Snap-to-object also has an incremental Snapshot Download capability embedded within it. When an initial snapshot from a source file system is restored into a new file system by downloading the snapshot from the object store, a further snapshot of that source file system can then be incrementally restored into the destination file system.

The outcome of using the snap-to-object feature is that the object store contains a full copy of the snapshot of the data, which can be used to restore the data on the original Hitachi Content Software for File cluster or onto another Hitachi Content Software for File cluster. The secondary cluster that mounts the snap-to-object snapshot does not need to be a mirror of the primary system. Any cluster size will work; however, performance is determined by the destination cluster size. This makes it ideal for cloud bursting. Consequently, the snap-to-object feature is useful for a range of use cases, such as:

Generic Use Cases (on-premises and cloud)

- **Data backup to HCP, HCP Cloud Scale, or cloud-based object store:** If a cluster fails beyond recovery because of an external event such as a fire, earthquake, flood, etc., the snapshot saved to the object store can be used to recreate the same data on another cluster or rehydrate onto the original cluster once recovered.
- **Data archival:** The periodic creation of data snapshots, followed by uploading the snapshot to an object store or the cloud to create an archive copy of the data.
- **Asynchronous mirroring of data:** Combining a Hitachi Content Software for File cluster with a replicated object store in another data center will mirror the data that can be mounted on a second cluster.

Cloud-Only Use Cases

- **Public cloud pause and restart:** Hitachi Content Software for File uses compute instances with local SSDs to create a cluster in the various cloud providers. For burst project-specific work, users may want to shut down or hibernate the cluster when no runs occur to save costs. The snapshot can be saved to an object store and rehydrated when needed again later.
- **Protection against single availability zone failure:** Utilizing the snap-to-object feature allows users to recover from an availability zone (AZ) failure. Should the first AZ fail, if the snapshot was replicated to a second AZ via the object store, it can be rehydrated in minutes by a cluster in the secondary AZ.

Hybrid Cloud Use Case

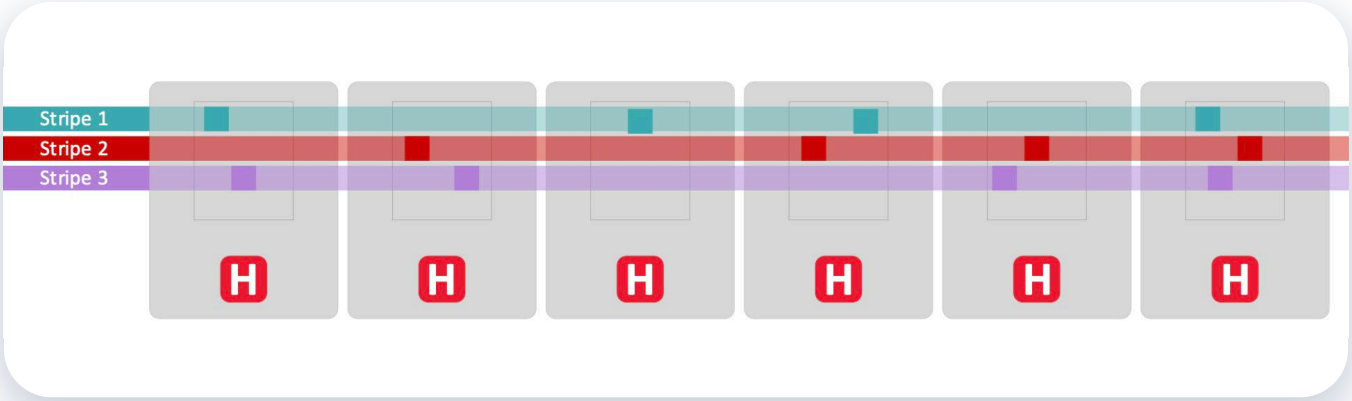
- **Cloud bursting:** An on-premises customer can benefit from cloud elasticity by using additional computational power for short periods. The file system can run in the cloud by uploading a snapshot to a cloud-based object store. After running in the cloud, the data can be deleted or archived, and the compute instances can be shut down.

Data Protection

Data protection is a critical function of any storage system, and challenges are amplified at scale. Without an appropriate internal data protection schema, file systems would need to be limited in size to accommodate the effects of disk or host rebuild time windows and minimize the risk of data exposure. Popular data protection schemes such as RAID6, internal replication (copies of blocks/files), and erasure coding are a compromise between scalability, protection, capacity, and performance.

With Hitachi Content Software for File, there is no data or metadata locality concept, as all data and metadata are distributed evenly across the storage nodes, improving scalability, aggregating performance, and improving resiliency. With the advent of high-speed networks, data locality contributes to performance and reliability issues by creating data hot spots and system scalability issues.

By directly managing data placement on the SSD layer, Hitachi Content Software for File can distribute the data across the storage cluster for optimal placement based on stripe sizes. Hitachi Content Software for File uses advanced algorithms to determine data layout; the placement of data matches the block sizes used by the underlying flash memory to improve performance. The minimum supported cluster size is 8, which allows for two full virtual spares for a rebuild from a 6+2 configuration. The bigger the cluster, the larger the stripe size that it can use, and the greater the storage efficiency and write performance the cluster gains per node.



Data Protection Schema

Hitachi Content Software for File manages protection so data is always safe and accessible:

- Configurable data protection levels from 4+2 to 16+4
- Distributed data protection schema
- Configurable failure domains
- End-to-end checksums for data integrity
- Metadata journaling
- Local snapshots and clones
- Snapshot-to-object for backup and DR

Hitachi Content Software for File uses failure domains to define data protection levels. Failure domains are fully configurable starting at the server host level, which provides single or multiple SSD level granularity. Data protection levels are flexible depending on the size and scale of the server cluster— the larger the cluster, the larger the recommended data stripe size for the best use of SSD capacity, improved performance, and higher resiliency. For granular protection, the data protection level is set at the cluster level, and parity can be set to two or four, meaning that the system can survive up to two or four simultaneous host failures without impacting data availability.

The data protection scheme follows the convention of data (N) + parity (2 or 4). An N + 2 protection level is sufficient for most production environments. An N+4 protection level is recommended for clusters with a large number (hundreds) of nodes due to application failures or lockups that can impact server availability.

In addition to core data protection, Hitachi Content Software for File recommends availability best practices such as servers with redundant power supplies, multiple NICs and switches for network redundancy, etc.

Virtual Hot Spare

A virtual hot spare is reserved capacity so that if a failure domain has failed, the system can undergo a complete data rebuild and still maintain the same net capacity. All failure domains always participate in storing the data, and the virtual spare capacity is evenly spread within all failure domains. The higher the virtual spare count, the more hardware is required to obtain the same net capacity.

The higher the hot spare count, the more relaxed the IT maintenance schedule for replacements. The virtual space is defined during the cluster formation and can be re-configured at any time. The default number of virtual spares is one.

Data Distribution

Hitachi Content Software for File uses a patented distributed data protection coding scheme that increases resiliency as the number of servers in the cluster scale. It delivers the scalability and durability of erasure coding without the performance penalty. Unlike legacy hardware and software RAID and other data protection schemes, WEKA's rebuild time gets faster and more resilient as the system scales because every server in the cluster participates in the rebuild process.

The cluster equally distributes data and metadata across logical buckets spaning failure domains (FD). A failure domain can be an individual storage node, a rack, or a data center. Hitachi Content Software for File can span availability zones (AZ) in cloud environments.

Unlike traditional hardware and software data protection schemes, Hitachi Content Software for File only places a single segment of a given data stripe inside any one server (or FD), so in the event of multiple drive failures within a single server, it will still be considered by a single failure of the domain. The data distribution mechanism always stripes across failure domains. The protection level is defined at the FD level. The cluster handles failures at the FD level, so individual or multiple failures within the FD are treated as a single failure. Depending on the resiliency chosen, data stripes are always spread across different server hosts, racks, or AZs. Hitachi Content Software for File's resiliency is user-configurable to define the number of failures to tolerate within a cluster to meet the application workload service level requirements.

When a failure occurs, the system considers the FD a single failure, regardless of how large the domain is defined. In addition to distributing stripes at the FD level, Hitachi Content Software for File also ensures a highly randomized data placement for improved performance and resiliency. As the cluster size grows, the probability of a hardware failure increases proportionally, but Hitachi Content Software for File overcomes this issue by distributing the stripes in a randomized manner. The more servers, the higher the number of random stripe combinations, making the probability of a double failure lower. Example: for a stripe size of 18 (16+2) and a cluster size of 20, the number of possible stripe combinations is 190; however, as the cluster size grows to 25, the number of possible stripe combinations is now 480,700. The number of possible stripe combinations is based on the following formula where C is the number of servers in a cluster, and S is the stripe size: C!/(S!*(C-S)!).

Hitachi Content Software for File Rebuilds

Hitachi Content Software for File uses several innovative strategies to return the system to a fully protected state as quickly as possible and be ready to handle a subsequent failure. This ensures that applications are not impacted by long data rebuild processes.

The cluster protects data at the file level, so it only needs to rebuild the data that is actively stored on the failed server or SSD. This means the rebuild times are faster than those of a traditional RAID solution or file server that protects data at the block layer. RAID controller-based systems typically rebuild all blocks on an affected storage device (SSD/HDD), including empty blocks, prolonging rebuilds and exposure time. Hitachi Content Software for File only needs to rebuild the specific file data affected by the failure. When a tiering-to-object policy is in place with a file system, a further benefit is that data that has already been tiered off to the object store is never impacted by a server failure because it is protected on the object store. In addition, any cached data (data tiered to object but remains on the flash tier until invalidated) does not need to be rebuilt either, limiting the rebuild priority to data that only resides on the flash tier.

Cluster stripes are comprised of 4k blocks. A stripe is distributed across all available failure domains. No two blocks belonging to the same stripe will be written to the same failure domain. Therefore, losing a failure domain results in only losing a single block from a stripe. All the remaining FDs in the cluster will participate in rebuilding any missing blocks in the stripe.

Examples include singular disk failures, host failures, or entire failure domain failures. Hitachi Content Software for File will rebuild data from that drive(s) or FD using a parity calculation and write that data across all remaining healthy FDs. This means that the larger the cluster size, the faster the rebuild and the more reliable the system becomes because more computing resources are available to participate in the rebuild process, and the stripes become more randomized across the hosts.

In the event of multiple failures, the system prioritizes data rebuilds, starting with data stripes in the least protected state. Hitachi Content Software for File looks for data stripes that are common to the failed hosts and rebuilds these data stripes first so the system can be returned to the next level higher of protection as fast as possible. This prioritized rebuild process continues until the system is returned to full redundancy. By contrast, in a replicated system, only the mirrored servers participate in the recovery process, impacting performance significantly. Erasure coding suffers from a similar problem, where only a small subset of the servers participates in the recovery. With Hitachi Content Software for File, the recovery rate is user-configurable, and the amount of network traffic dedicated to rebuilding can be changed at any time, so administrators have complete control to determine the best tradeoff between continued application performance and time-to-recovery.

Power-Failureand End-to-End Data Protection

Using a checksum process to ensure data consistency, Hitachi Content Software for File provides end-to-end data protection for both reads and writes. Checksums are created on write and validated on reads. The cluster always stores data and checksum information separately from each other on different physical media for improved protection.

Hitachi Content Software for File provides additional data integrity capabilities by protecting against data loss due to power failures. When a write is acknowledged back to the client, it is safely protected from server failures or data-center-wide power failure through a journaling process. The innovative data layout and algorithms enable it to recover from a data-center-wide power failure in minutes because there is no need to do a complete file system consistency check (FSCK). For most other file systems, the FSCK process recovery time is proportional to the size of the recovered file system. In large-scale deployments, this recovery can take days or weeks.

Automated Data Rebalancing

WekaFS proactively monitors and manages a WEKA cluster's performance, resiliency, and capacity health status. This allows the system to calculate the utilization levels (performance and capacity) of hosts to redistribute data automatically and transparently across the cluster to prevent hot spots.

The benefit is that Hitachi Content Software for File can maintain well-balanced cluster performance and data protection as capacity and usage change. Another advantage is that as additional SSDs are added to existing servers or the cluster is expanded with more servers, the cluster automatically rebalances to enhance performance, resiliency, and capacity without requiring costly downtime for data migration. Matched capacity SSDs are not required, which allows you to leverage new technology and save money as SSD prices decline.

Container Storage Integration

Container Storage Interface (CSI) is a standard developed to provision and manage shared file storage for containerized workloads. The CSI Plugin for Kubernetes provides an interface between the logical volumes in a Kubernetes environment (Persistent Volumes (PVs)) and the storage, enabling customers to deploy stateless clients to connect storage to the appropriate container. The CSI plugin provisions a Kubernetes pod volume either via PV (by an administrator), or it can be dynamically provisioned via a Persistent Volume Claim (PVC). This feature simplifies moving containerized workloads to the cloud or sharing data across multiple Kubernetes clusters. The CSI plugin also supports using quotas to help manage space consumption for containerized applications.

Multi-Tenant Organizations

Hitachi Content Software for File supports the construct of organizations, an element of multi-tenancy that offers hierarchical access controls. Access can be separated so that data is managed at the organizational level and only visible to that group's members. Hitachi Content Software for File can support up to 64 organizations. Within an organization, logical entities obtaining control of that data are managed by the Organization Administrator, not the Cluster Administrator. Cluster Administrators can create organizations, define the Organizational Admin, delete organizations, and monitor capacity usage by the organization file system.

Capacity Quotas

As noted, there are many ways that Hitachi Content Software for File manages capacity utilization across organizations.

- Organizational-level quotas allow groups to manage their own file systems and capacity. Hitachi Content Software for File supports up to 64 organizations.
- File system level capacity allows different projects or departments to have their own allocated capacity. The file system supports up to 1024 file systems on a single storage namespace.
- Directory-level quotas provide a quota per project directory, which is useful when there are many projects within a single file system. Quotas can be set at an advisory level, as hard quotas or as soft quotas.

Quality of Service

The Hitachi Content Software for File clients also has additional performance management capabilities in the form of QoS functionality. This is a limiting function where you can set both a preferred throughput and maximum throughput. The client will attempt to limit as close to the value of the preferred performance as possible but allow bursting up to the maximum amount if resources are available. This enables per-application performance management when accessing the file system. Combine quotas and organizational controls allows fine-grained resource management within the Hitachi Content Software for File system.

Authentication and Access Control

Hitachi Content Software for File provides authentication services at the user and client-server levels to validate that the user or client can view and access data. The cluster allows different authenticated mount modes, such as read-only or read-write, and is defined at the file system level. Authenticated mounts are defined on the organizational level and are encrypted by an encryption key. Only clients with the proper keys can access authenticated mount points. This methodology increases security by drastically limiting access to specific subsets of an organization and limiting access to clients with the proper encryption key. Hitachi Content Software for File supports the following:

- LDAP (Lightweight Directory Access Protocol) is a networking protocol that provides directory services across many different platforms.
- Active Directory, a Microsoft implementation of LDAP, is a directory service that can store information about network resources. It is primarily used to authenticate users and groups who want to join the cluster.
- Role-Based Access Control (RBAC) delivers different privileges to users and administrators. Some users can be granted full access rights, while others have read-only rights.

Role-Based Access Controls

Every Hitachi Content Software for File system user has one of the following defined roles:

Cluster Admin: A user with additional privileges over regular users. These include the ability to:

- Create new users
- Delete existing users
- Change user passwords
- Set user roles
- Manage LDAP configurations
- Manage organizations

Additionally, the following restrictions are implemented for Cluster Admin users to avoid situations where a Cluster Admin loses access to a WEKA system cluster:

- Cluster Admins cannot delete themselves
- Cluster Admins cannot change their role to a regular user role

Organization Admin: A user with privileges similar to those of Cluster Admins, except that these privileges are limited to the organization level. They can perform the following within their organization:

- Create new users
- Delete existing users
- Change user passwords
- Set user roles
- Manage the organization's LDAP configuration

Furthermore, to avoid situations where an organization admin loses access to a WEKA system cluster, the following restrictions are implemented for Organization Admins:

- Organizational Admins cannot delete themselves
- Organizational Admins cannot change their role to a regular user role

Regular: A user with read and write privileges. A user that should only be able to mount file systems and:

- Can log in to obtain an access token
- Can change their password
- Cannot access the UI or run other CLI/API command

Read-only: A user with read-only privileges

S3: A user to run S3 commands and APIs. This user can operate within the limits of the S3 IAM policy attached to it.

Encryption In-Flight and At-Rest

Hitachi Content Software for File provides complete end-to-end encryption from the client to the storage hardware nodes, making it the most robust encrypted file system commercially available. Encryption is set at several levels.

- File system level: When the file system is created, some file systems that are deemed critical can be encrypted, while others are not.
- Hardware level, with Storage Encrypted Drives, provides node-wide encryption at rest.

When files are encrypted, the data will remain encrypted even if cold blocks underlying the file are sent to an object store tier. Hitachi Content Software for File's encryption solution protects against physical media theft, low-level firmware hacking on the storage media, and in-line encryption to protect MitM.

Hitachi has demonstrated that encrypted file systems have minimal impact on application performance when using the client.

Key Rotation and Key Management

Hitachi Content Software for File supports any key management system (KMS) that is compliant with KMIP (the Key Management Interoperability Protocol) 1.2 and above, as well as Hashicorp Vault's proprietary API. Cluster keys are rotated by the KMS, file system keys can be rotated via the KMS and re-encrypted with the new KMS master key, and file keys can be rotated by copying the file. Hardware encryption keys are also managed through the KMS.

File data on the object store is also encrypted. When uploading a snap-to-object to the object store, among other file system parameters, the file system key is included and encrypted with a unique "backup-specific" cluster key available via the KMS and used for all snap-to-object backups and restores. When Hitachi Content Software for File pushes a snapshot to an object store, the data is fully protected and can be authenticated only through the KMS system.

Flexible Deployment Options (On-Premises and Cloud)

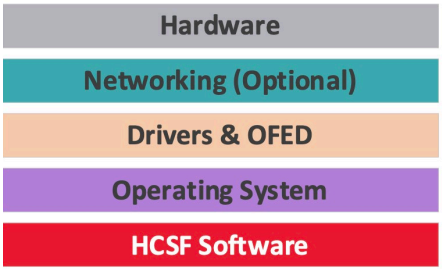
Whether your applications run on bare metal for performance, in a virtual or containerized environment for ease of deployment, or in the public cloud for on-demand scalability, Hitachi Content Software for File is a single storage solution that allows you to choose the environment best suited for your application based on performance, scale, and economics. Clients can support bare metal, virtualized, containerized, and cloud environments.

Hitachi Content Software for File can scale to thousands, starting at eight nodes. As infrastructure ages, it enables cluster expansion with new nodes and then seamlessly retires older generation nodes. This allows lifecycle management of hardware refreshes without the need to perform lift-and-shift data migration to new storage.

Hitachi Content Software for File can be deployed in a few ways.

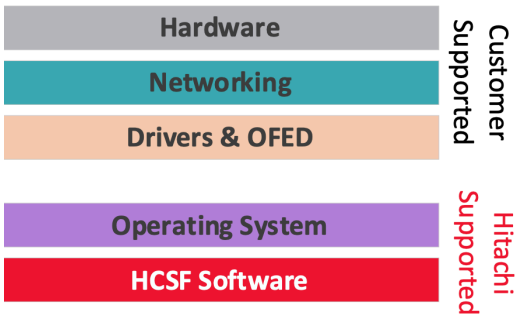
Dedicated Storage Nodes

Hitachi Content Software for File can be deployed as a dedicated storage node from Hitachi, in which all the system's resources are dedicated to storage services while applications run on a separate compute infrastructure. This deployment style provides all hardware, software, and OS components predefined and configured for ease of deployment and support.



Software Storage Appliances

Hitachi Content Software for File can be deployed as a software-only appliance from Hitachi on any validated and supported third-party hardware platform. Hitachi supports the OS and Hitachi Content Software for File software, allowing the reuse of existing hardware. All customer-supported parts must be in the current support matrix for the release. Supported hardware, OS, and OFED levels can be found in the release notes for each software release.



In both deployment methodologies, Hitachi Professional services are provided to ensure proper setup of the public cloud.

Public Cloud

Hitachi Content Software for File can be deployed on major public hyperscalers, and the deployment process is similar to that of the Software Storage Appliance. You choose a server configuration (EC2, OCI Compute Shape, GCP Compute Engine, etc.), ensure that NVMe devices are in or attached to the compute resource, assign networking, and then deploy the cluster.

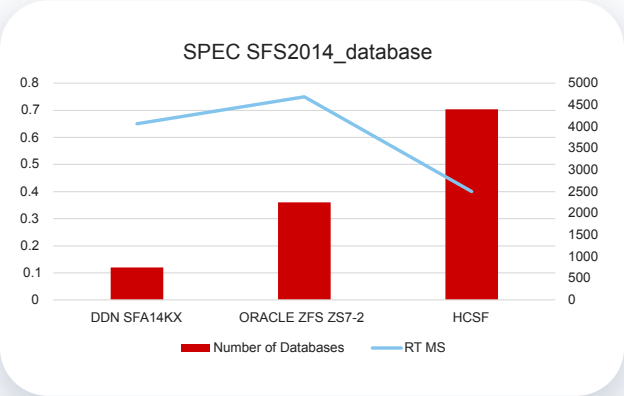
Performance in the cloud is coupled to the number of NVMe devices in a cluster and network speeds. For best performance, please work with a Hitachi system engineer to get a sizing recommendation.

Hitachi Content Software for File PERFORMANCE PROOF POINTS

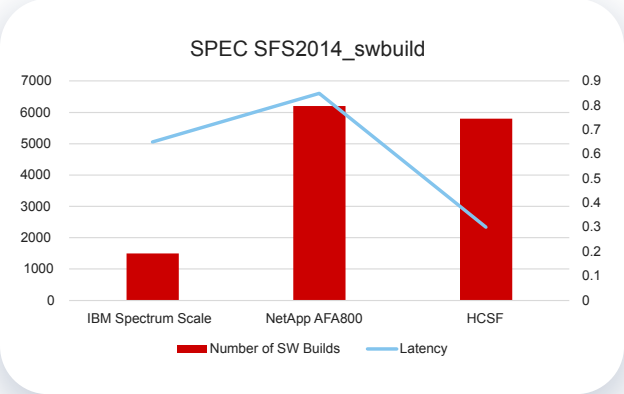
Traditionally, high-performance goals for file-based applications were achieved by running them on a local file system utilizing Direct Attached Storage (DAS), provisioning all-flash capacity over fabric-based NVMe (NVMeoFC), or by using parallel file systems with protocols such as SMB and NFS to share with multiple clients. The problem with these approaches is that they do not scale with multiple clients (DAS) or have performance limitations of Linux NFS or Windows SMB. Hitachi Content Software for File delivers significant performance benefits over this approach utilizing a shared POSIX-compliant file system. This approach provides the local caching advantages of DAS and the shareability advantages of NFS or SMB. As a result, Hitachi Content Software for File allows you to run any workload on a networked shared file system with as much or more performance than DAS can offer. This, coupled with the extremely low latency of 4K I/O operations to the application, makes using AFA volumes for a local file system obsolete.

Standard Performance Evaluation Corporation: SPEC.org

The SPEC SFS® 2014 benchmark is a version of the Standard Performance Evaluation Corporation benchmark suite measuring file server throughput and response time, providing a standardized method for comparing performance across different vendor platforms. WekaFS holds the leadership position for databases, Virtual Desktop Infrastructure (VDI), Electronic Design Automation (EDA), and Video Data Acquisition (VDA), with a close second for software builds. The following demonstrates Hitachi Content Software for File's performance for databases. WekaFS supported 4480 databases, 2x more than the next closest submission, at a 340-microsecond overall response time.

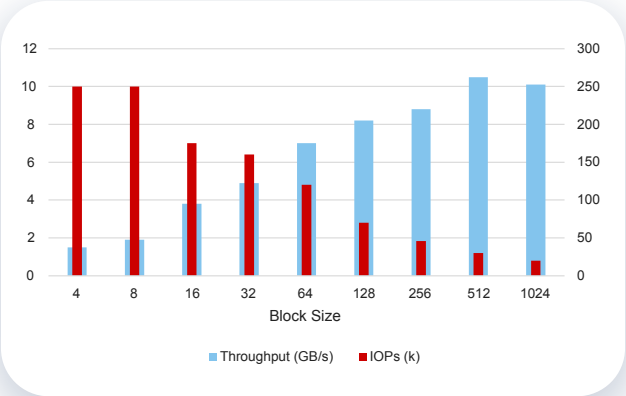


The following demonstrates the low latency of Hitachi Content Software for File. While the NetApp AFF A800 supported 8% more builds than WEKA, the latency for WekaFS was 260 microseconds compared to 830 microseconds for NetApp. In other words, WekaFS can complete over 3x as many software builds as NetApp at the same time.

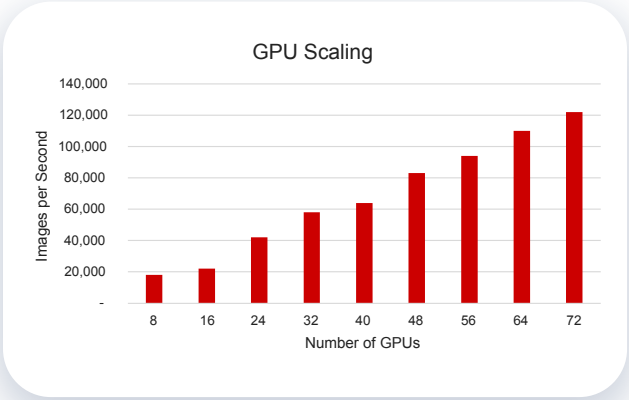


Performance to GPU Storage

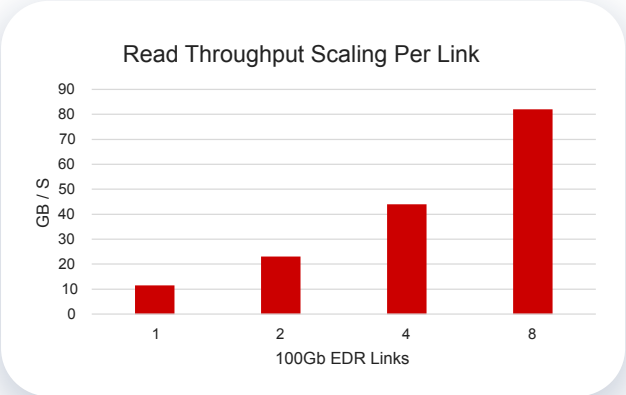
Hitachi Content Software for File is an ideal solution for GPU-intensive workloads. Hitachi Content Software for File has developed a reference architecture for the NVIDIA® DGX-1™ GPU System. FIO testing provides a baseline measure of the I/O capability of the reference architecture. A performance test was conducted with a single DGX-1 system to establish the performance that could be delivered with the minimum hardware configuration on a single 100-Gbit InfiniBand link to the host. The following shows that Hitachi Content Software for File can fully saturate a 100-Gbit link, delivering a peak read performance of 10.8 GBytes/second to a single DGX-1 system. The IOPs performance measurement shows that Hitachi Content Software for File delivered over 250,000 IOPs to a single DGX-1 system on one 100-Gbit network link.



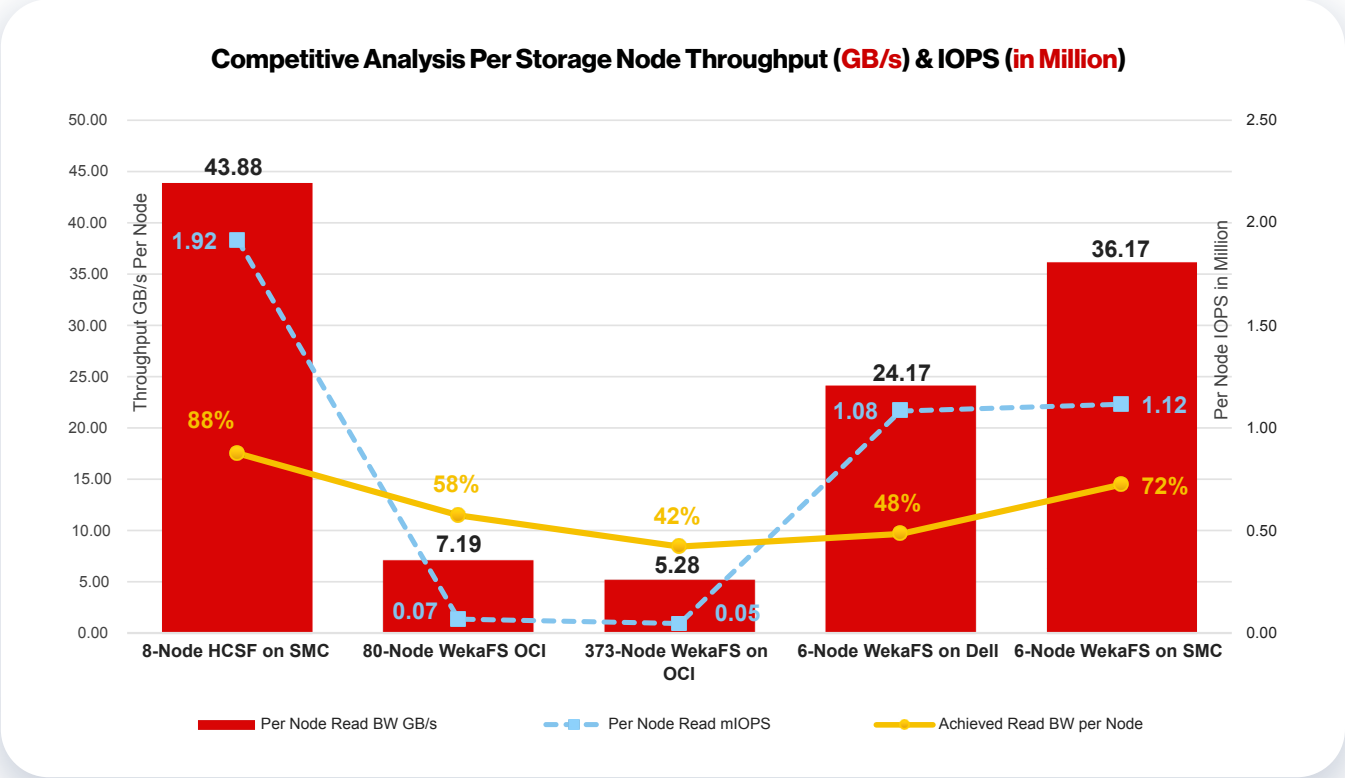
The next set of performance testing measures Hitachi Content Software for File's ability to scale from a single GPU server to multiple GPU servers. The following demonstrates the solution's ability to maintain linear scaling from 1 NVIDIA DGX-1 to 9 NVIDIA DGX-1 systems. Each NVIDIA DGX-1 had 8 GPUs, for a total scaling to 72 Tesla V100 GPUs.



The final set of performance tests measures Hitachi Content Software for File's ability to deliver performance to a single GPU server utilizing GPUDirect Storage. Hitachi Content Software for File scaled performance inside a single NVIDIA DGX-2™ GPU server from one 100Gbit EDR link to 8 EDR links with linear scaling, saturating the network bandwidth.



When comparing Hitachi's Hitachi Content Software for File offering against other WekaFS-based offerings, Hitachi can demonstrate a 71% improvement in small file random read I/O and a 21% improvement in large block read bandwidth compared to similar offerings.



About Hitachi Vantara

Hitachi Vantara is transforming the way data fuels innovation. A wholly owned subsidiary of Hitachi, Ltd., we're the data foundation the world's leading innovators rely on. Through data storage, infrastructure systems, cloud management and digital expertise, we build the foundation for sustainable business growth.

Hitachi Vantara
A Hitachi Group Company

Corporate Headquarters
2535 Augustine Drive
Santa Clara, CA 95054 USA
hitachivantara.com | community.hitachivantara.com

Contact Information
USA: 1-800-446-0744
Global: 1-858-547-4526
hitachivantara.com/contact

HV-BTD-WP-Hitachi-Content-Software-for-File-19July24-A